



**REPÚBLICA DE CUBA  
MINISTERIO DE EDUCACIÓN SUPERIOR  
UNIVERSIDAD DE GRANMA  
FACULTAD DE INFORMÁTICA**



**UNIVERSIDAD TÉCNICA DE COTOPAXI  
UNIDAD ACADÉMICA DE CIENCIAS DE LA INGENIERÍA Y  
APLICADAS**

**SISTEMA DE GESTIÓN DE ACTIVOS FIJOS TANGIBLES DEL  
INSTITUTO DE SUELOS DE GRANMA**

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO  
DE INGENIERO EN INFORMÁTICA**

**Autores:** Marco Vinicio Naranjo Gallo

Eddison Tomás Lema Chuqui

**Tutores:** Ing. Ibet de los Ángeles Pascual Sánchez

Ing. Ramón Osmany Ramírez Tasé

Lic. Rony González Vázquez

**Junio, 2013**

## **AGRADECIMIENTOS**

**Eddison Lema**

- ❖ *A Dios por la vida que me dio y la fortaleza que imprimió en mí persona, y permitiendo sabiduría llegaran a mí.*
- ❖ *A mí, madre que con su amor y dedicación se entregó por completo a mis estudios, apoyándome en la buena y en las malas*
- ❖ *A mí padre por todo el apoyo, sus consejos, y sus incentivaciones que estudie y nunca me decaiga.*
- ❖ *A mi hermano Wilmer, es lo mejor que tengo en la vida ya que es como si fuera mi hijo y lo quiero muchísimo.*
- ❖ *A mis hermanas por el apoyo incondicional que me han brindado por ser como son alegres y alégame la vida en los difíciles momentos de mi vida estudiantil.*
- ❖ *A mis tutores por toda la ayuda que me han brindado, por sus orientaciones certeras y su paciencia. En especial Ing. Ramón Ramírez*
- ❖ *A la universidad técnica de Cotopaxi por haberme abierto la puertas para hacia un mundo de conocimiento y cumplir una meta más en mi vida.*
- ❖ *A todos los que de una u otra forma han influido en mi formación profesional.*

**Gracias eternamente.**

**DEDICATORIA**  
**Eddison Lema**

❖ *A quién le debo todo lo bello y bueno que hay en mi vida.*

*A dios.*

❖ *Para la mujer que me dio la vida, me inculco con sus buenas enseñanzas y medio todo su amor incondicional, a mi madre*

*Carmen Chuqui Jaya*

❖ *A mi padre, por ser mas que un padre un amigo que me aconsejaba en todo momento, y su cariño.*

*Tomás Lema de la Cruz.*

❖ *A mi hermano chiquito por las alegrías que me ha dado.*

*Wilmer Lema.*

❖ *A mis 5 hermanos, que ser las mejores hermanas y mujeres que he tenido.*

*Todos por igual.*

❖ *A mi tía, primos y primas por los consejos que me dieron durante mi vida estudiantil y como lo hacen hasta hora.*

***Por eso este trabajo es dedicado para ustedes....***

## *DEDICATORIA*

*Marco Naranjo*

### *A Dios.*

*Por haberme permitido llegar hasta este punto y haberme dado salud para lograr mis objetivos, además de su infinita bondad y amor.*

### *A mi Esposa e Hijo.*

*Anabel Espin y Mathias Naranjo, gracias por el valor y las fuerzas para seguir luchando por alcanzar las metas propuestas y por la comprensión de su parte para poder alejarme de ustedes los amo.*

### *A mi Madre Moníca Patricia Gallo Mora*

*Por darme la vida, el apoyo incondicional día a día a pesar de la distancia que nos separa, por confiar en mí y darme una carrera para mi futuro te amo Madrecita.*

### *A mi Padre Washington Vinicio Naranjo Vega*

*Por apoyarme siempre y por estar junto a mí por sus consejos por su apoyo incondicional y por confiar en mí desde un inicio de mi carrera te amo querido Padre.*

### *A mi Familia*

*Mis abuelitos Alfonso (QEPD), Carmen Gallo y Evangelina Vega, mis hermanas Evelyn, Yajaira, Carol, mis tíos Helen, Edy, Marlon, Mercedes, Blanca, Alfonso, a todos mis primos, por apoyarme siempre en las buenas y malas situaciones, por los consejos sabios que siempre me han brindado para ser una persona correcta y lograr alcanzar el objetivo que ahora lo estoy consiguiendo.*

*Este trabajo es para ustedes....*

## *AGRADECIMIENTOS*

*Marco Naranjo*

*A Dios, por estar conmigo en todos los momentos de mi vida.*

*A mi esposa e hijo, por darme valor y fuerzas todos los días para luchar por ustedes, por tu apoyo incondicional y por estar siempre conmigo en las buenas y las malas te amo Anabel Espín.*

*A mi madre, por estar siempre conmigo a pesar de no vivir juntos te amo madre.*

*A mi padre, por estar conmigo en toda mi carrera universitaria.*

*A mis Hermanas Evelyn, Yajaira, Carol, por alegrarme la vida por todos los momentos de felicidad compartidos juntos, las amo mis enanitas preciosas.*

*A mi abuelita Evangelina Vega, por tenerme a su lado desde que inicie mis estudios y guiarme siempre por la senda de la humildad y la bondad.*

*A mi abuelita Carmen Gallo, por estar siempre guiándome y educándome para ser una persona respetuosa y entregada a Dios.*

*A mis tíos Marlon, Helen y Edy por estar siempre cuando más los necesito, con su apoyo incondicional siempre sus consejos y su confianza depositada en mí para lograr este triunfo tan importante en mi vida.*

*A mis tíos Rene Ortiz y Mercedes Naranjo, por ser mis segundos padres y abrirme siempre las puertas de su hogar y de su corazón, por tenerme como un hijo más y por todo el apoyo que me han brindado.*

*A mis suegros Mario Espín y Teresa Balseca, por darme la confianza para formar parte de su familia y por su apoyo incondicional para lograr alcanzar nuestras metas propuestas.*

*A mis tutores Ing. Ramón Ramírez e Ing. Ibet Pascual, por brindarme sus conocimientos y estar siempre pendiente del trabajo realizado, por los buenos momentos compartidos y la confianza, para culminar con el trabajo de investigación.*

*A todos los que de una u otra forma han influido en mi formación profesional.*

*Gracias eternamente.*

## Resumen

En el Departamento de Economía del Instituto de Suelos de Granma se lleva el control de los activos fijos tangibles con los que cuenta la institución, toda la información correspondiente a los mismos se gestiona de forma manual y mediante el tabulador electrónico de *Microsoft Excel*, lo cual ocasiona que la información se encuentra dispersa, se requiere de un gran número de horas de trabajo para su actualización, es sensible a pérdidas y deterioro.

Por lo planteado anteriormente, en la presente investigación se desarrolla un sistema de entorno web con nuevas potencialidades, para facilitar la gestión de la información concerniente a los activos fijos tangibles del Instituto de Suelos de Granma, permitiendo una mayor consistencia y seguridad de la información almacenada, facilitando el manejo y el rápido acceso a la misma. En este documento se expone la fundamentación teórica, las herramientas utilizadas, las características del sistema, diseño, implementación y las pruebas realizadas. El desarrollo de este trabajo está basado en tecnologías multiplataforma; mediante el uso de lenguajes de programación como el HTML, JavaScript y PHP, así como Hojas de Estilos (CSS). Como servidor web se utilizó el Apache y como gestor de bases de datos el MySQL, para el crear y editar las páginas se recurrió al *Dreamweaver CS5*, además como metodología de desarrollo *Extreme Programming (XP)*.

### **Summary**

In the Department of Economy of the Institute of Floors of Granma the control of the fixed tangible assets is taken with those that it counts the institution, all the information corresponding to the same ones it was taken in a manual way and by means of Microsoft Excel, that which caused that the information was dispersed, was required of a great number of hours to be upgraded, it was sensitive to losses and deterioration.

For that outlined previously, it is that in the present investigation a system of environment web is developed with new potentialities, to facilitate the management from the concerning information to the fixed tangible assets of the Institute of Floors of Granma, allowing a bigger consistency and security of the stored information, facilitating the handling and the quick access to the same one. In this document it is exposed the theoretical foundation, the used tools, the characteristics of the system, design, implementation and the carried out tests. The development of this work is based on technologies multiplatform; it was made use of programming languages like HTML, JavaScript and PHP, as well as Leaves of Styles (CSS). As servant web the Apache was used and I eat agent of databases MySQL, for create and publish the pages it was appealed Dreamweaver CS5, also as development methodology Extreme Programming (XP).

## Índice

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO 1. FUNDAMENTOS PARA LA GESTIÓN Y CONTROL DE LOS ACTIVOS FIJOS TANGIBLES.....</b>	<b>6</b>
<b>1.1 INTRODUCCIÓN.....</b>	<b>6</b>
<b>1.2 PROCESO DE GESTIÓN DE ACTIVOS FIJOS TANGIBLES.....</b>	<b>6</b>
<b>1.2.1 Gestión de activos fijos tangibles en el sector empresarial estatal cubano .....</b>	<b>6</b>
<b>1.3 SISTEMAS INFORMÁTICOS PARA LA GESTIÓN DE ACTIVOS FIJOS TANGIBLES ..</b>	<b>8</b>
<b>1.4 HERRAMIENTAS, TECNOLOGÍAS Y METODOLOGÍAS PARA EL DESARROLLO DE SISTEMAS INFORMÁTICOS.....</b>	<b>11</b>
<b>1.4.1 Aplicaciones web .....</b>	<b>11</b>
<b>1.4.2 Lenguajes y librerías para el desarrollo de aplicaciones web .</b>	<b>16</b>
<b>1.4.3 Servidores web.....</b>	<b>20</b>
<b>1.4.4 Sistemas de administración de bases de datos.....</b>	<b>22</b>
<b>1.5 METODOLOGÍAS DE DESARROLLO DE SOFTWARE.....</b>	<b>25</b>
<b>1.5.1 Proceso Unificado de desarrollo de software (RUP) .....</b>	<b>25</b>
<b>1.5.2 Programación Extrema (XP).....</b>	<b>26</b>
<b>1.6 JUSTIFICACIÓN DE LAS HERRAMIENTAS SELECCIONADAS.....</b>	<b>27</b>
<b>1.7 CONCLUSIONES DEL CAPÍTULO .....</b>	<b>28</b>
<b>CAPÍTULO 2 DESCRIPCIÓN, CONSTRUCCIÓN Y VALORACIÓN DE LA PROPUESTA DE SOLUCIÓN .....</b>	<b>29</b>
<b>2.1 INTRODUCCIÓN.....</b>	<b>29</b>
<b>2.2 PERSONAS RELACIONADAS CON EL SISTEMA.....</b>	<b>29</b>
<b>2.3 FASE DE EXPLORACIÓN.....</b>	<b>30</b>
<b>2.3.1 Historia de usuario .....</b>	<b>30</b>
<b>2.3.2 Valoración de sostenibilidad.....</b>	<b>32</b>
<b>2.4 FASE DE PLANIFICACIÓN.....</b>	<b>36</b>
<b>2.4.1 Estimación de esfuerzos por historias de usuarios.....</b>	<b>36</b>
<b>2.4.2 Plan de duración de las iteraciones .....</b>	<b>38</b>
<b>2.4.3 Plan de entrega.....</b>	<b>39</b>
<b>2.5 ITERACIONES .....</b>	<b>40</b>



2.5.1 Tareas .....	40
2.6 PRODUCCIÓN .....	43
2.6.1 Diseño .....	44
2.6.2 Codificación .....	45
2.6.3 Pruebas .....	45
2.7 MANTENIMIENTO .....	52
2.8 MUERTE DEL PROYECTO .....	53
2.9 CONCLUSIONES DEL CAPÍTULO .....	53
<b>CONCLUSIONES GENERALES .....</b>	<b>54</b>
<b>RECOMENDACIONES .....</b>	<b>55</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>56</b>

## Índice de tablas

<b>Tabla 2.1</b>	Personas relacionadas con el sistema.....	29
<b>Tabla 2.2</b>	Historia de usuario “Gestionar áreas” . .....	30
<b>Tabla 2.3</b>	Historia de usuario “Gestionar activos fijos tangibles”.....	31
<b>Tabla 2.4</b>	Historia de usuario “Gestionar control de servicios”.....	31
<b>Tabla 2.5</b>	Historia de usuario “Gestionar movimiento” . .....	32
<b>Tabla 2.6</b>	Estimación de esfuerzo por historia de usuario.....	36
<b>Tabla 2.7</b>	Plan de duración de las iteraciones. ....	38
<b>Tabla 2.8</b>	Plan de entrega por iteraciones. ....	39
<b>Tabla 2.9</b>	Tarea insertar activos fijos tangibles. ....	40
<b>Tabla 2.10</b>	Tarea modificar activos fijos tangibles.....	41
<b>Tabla 2.11</b>	Tarea eliminar activos fijos tangibles. ....	41
<b>Tabla 2.12</b>	Tarea visualizar activos fijos tangibles .....	41
<b>Tabla 2.13</b>	Tarea insertar área.....	42
<b>Tabla 2.14</b>	Tarea modificar área. ....	42
<b>Tabla 2.16</b>	Tarea visualizar área.....	43
<b>Tabla 2.17</b>	Caso de prueba insertar correctamente activos fijos tangibles.....	45
<b>Tabla 2.18</b>	Caso de prueba error al insertar activos fijos tangibles. ....	46
<b>Tabla 2.19</b>	Caso de prueba modificar activos fijos tangibles.....	46
<b>Tabla 2.20</b>	Caso de prueba error al modificar activos fijos tangibles. ....	47
<b>Tabla 2.21</b>	Caso de prueba eliminar activos fijos tangibles correctamente.....	48
<b>Tabla 2.22</b>	Caso de prueba error al eliminar activos fijos tangibles. ....	48
<b>Tabla 2.23</b>	Caso de prueba insertar correctamente una área. ....	49
<b>Tabla 2.24</b>	Caso de prueba error al insertar una área. ....	49
<b>Tabla 2.25</b>	Caso de prueba modificar correctamente una área. ....	50
<b>Tabla 2.26</b>	Caso de prueba error al modificar una área.....	50
<b>Tabla 2.27</b>	Caso de prueba eliminar una área correctamente. ....	51
<b>Tabla 2.28</b>	Caso de prueba error al eliminar una área.....	51
<b>Tabla 2.29</b>	Caso de prueba visualizar áreas correctamente. ....	52

### **Introducción**

El mundo contemporáneo exige que cada día exista una mejor gestión de los recursos financieros que posea cualquier organización, de forma que la triada del triunfo financiero (eficiencia, eficacia y calidad) generen la riqueza que satisfaga las necesidades crecientes del pueblo. En el caso muy particular de Cuba, se preocupa y ocupa por el perfeccionamiento de su modelo económico, el cual demanda que los organismos implicados de la fiscalización y control estén debidamente capacitados, para que puedan realizar su labor y que reine una sociedad sin corrupción e ilegalidades.

La Dra. Cabrera Elejarde. (2009) define la cultura económica “Como el sistema complejo de interacciones sociales que se establecen en el proceso de producción de bienes materiales y servicios en un contexto histórico determinado que trasciende a toda actividad humana, se revela a través de los conocimientos, los modos de actuación, la conciencia económica incluye las habilidades y destrezas adquiridas en el proceso de desarrollo y en su funcionamiento humano y se expresa en la creación y conservación de valores materiales en estrecha relación con los valores éticos–económicos y espirituales en general” (de Armas Phillips y Machado Barrios, 2010).

La cultura económica y el desempeño profesional en el nuevo modelo de perfeccionamiento de la función y de dirección del sistema de control interno involucra a todas las áreas de las empresas y de hecho involucra a todos los directivos, administrativos y trabajadores, es la dimensión cultural del desarrollo, es fundamento de todo ese proceso, la ciencia y la cultura son formas de adquirir conocimientos.

El tema de los Activos Fijos Tangibles, sobre todo, es muy común en cualquier empresa. El tratamiento de la adquisición; del cálculo y registro de la depreciación; los mantenimientos y reparaciones corrientes y capitalizables; las bajas por inservibles, ventas o intercambios; así como las reglas de valoración y exposición en los estados financieros constituyen los núcleos básicos e imprescindibles en el estudio del tema, que a su vez es vital para el funcionamiento de cualquier institución.

En busca de mejores y más prácticas soluciones, las empresas a nivel mundial han adoptado nuevas medidas, que van desde la reestructuración empresarial hasta la importación de nuevas tecnologías. El uso de las Tecnologías de la Información y las Comunicaciones (TIC) unido a los servicios Web aporta numerosas ventajas que vienen dadas de acuerdo a sus características fundamentales, como agilizar los procesos, mejorar la productividad, aumentar la eficacia, permite la interactividad, instantaneidad, innovación, digitalización, interconexión y diversidad (Mora Guevara y López Escala, 2012).

El proceso de informatización de las empresas cubanas ha planteado el desarrollo de sistemas informáticos que faciliten la gestión de información, así como la informatización de procesos para lograr un mejor funcionamiento y rendimiento de estas instituciones de acuerdo a las necesidades de la actualidad. Entre los procesos preponderantes por la importancia que se le otorga cuenta la gestión y control de los recursos financieros, entre los que se encuentran los activos fijos tangibles.

El Instituto de Suelos de Granma no se queda atrás en este proceso de informatización, esta es la institución estatal encargada de garantizar el uso, manejo y conservación de los suelos de la provincia, para lograr una mayor eficiencia en sus procesos es necesario garantizar un eficiente control, uso y aprovechamiento de sus recursos materiales, para lo cual es inminente la aplicación de las TICs.

En el Instituto de Suelos de Granma, el Departamento de Economía es el encargado de controlar los activos fijos tangibles de la institución. En ocasiones a la entidad le realizan inspecciones y auditorías por parte de las empresas a las que se subordina, en cada inspección o auditoría, el Instituto de suelos debe presentar el estado de su control interno mensual que es un valor representativo al 10% de los activos fijos tangibles, esta información se controla a través de un sistema de modelos de control.

Todo este proceso se realiza de forma manual presentando algunas limitantes:

- Una parte del proceso se realiza mediante el tabulador electrónico de *Microsoft Excel* ocasionando que la información se encuentre dispersa entre documentos impresos y electrónicos.

- Para mantener actualizada la información se requiere de un gran número de horas de trabajo, debido al gran volumen proveniente de fuentes diversas y dispersas.
- Para conocer el estado de los medios básicos es necesario consultar informes aislados que en ocasiones contienen errores, pueden sufrir deterioro y están sensibles a pérdidas.
- No existe seguridad de acceso a los documentos, estos pueden ser modificados o sustraídos, lo que atenta contra la validez y veracidad de la información.
- Se requiere de gran cantidad de papeles para el control de la información.

Luego de analizar la situación existente en el Departamento de Economía del Instituto de Suelos de Granma, surge el siguiente **problema de investigación**: ¿Cómo favorecer el proceso de gestión de la información relacionada con los activos fijos tangibles del Instituto de Suelos de Granma?

Teniendo como **objeto de estudio**: Proceso de gestión de la información de los activos fijos tangibles del Instituto de Suelos de Granma.

Enmarcado en el siguiente **campo de acción**: Sistema de gestión de información de los activos fijos tangibles del Instituto de Suelos de Granma.

Para guiar la investigación se propone la siguiente **idea a defender**: Con el desarrollo de un sistema informático, se contribuirá a mejorar la gestión de información de los activos fijos tangibles del Instituto de Suelos de Granma.

Para contribuir a resolver el problema planteado se propone el siguiente **objetivo general**: Desarrollar un sistema informático para la gestión de la información de los activos fijos tangibles del Instituto de Suelos de Granma.

### **Objetivos específicos:**

- Caracterizar el proceso de gestión de la información de los activos fijos tangibles del Instituto de Suelos de Granma.
- Desarrollar un sistema informático para la gestión de la información de los activos fijos tangibles del Instituto de Suelos de Granma.

### Tareas

- Revisar bibliografía científica referente al proceso de control de activos fijos tangibles.
- Analizar el proceso de gestión de los activos fijos tangibles del Instituto de Suelos de Granma.
- Analizar las tendencias actuales de las metodologías, tecnologías y herramientas a utilizar en el desarrollo de sistemas informáticos.
- Diseñar y normalizar la base de datos que mantenga la información persistente.
- Implementar un sistema informático para la gestión de la información referente a los activos fijos tangibles del Instituto de Suelos de Granma.

### Métodos Científicos de Investigación

Para dar cumplimiento a las tareas anteriores se emplearon los siguientes métodos y técnicas.

#### Métodos teóricos:

**Análisis y síntesis:** Para el desarrollo del sistema se realizó una investigación previa de los procesos que intervienen en el desarrollo de software y los principales elementos que integran las metodologías.

**Histórico – lógico:** se utilizó para el estudio del desarrollo histórico del proceso de control de los activos fijos tangibles y de los antecedentes de investigaciones relacionadas con la gestión de información.

**Modelación sistémica:** permitió realizar un análisis del objeto de la investigación y su posterior modelación, usando las herramientas informáticas de Ingeniería de Software y las técnicas adecuadas de programación de sistemas.

**Empírico:** se utilizó la observación.

Observación: se utilizó para acercarse al trabajo de los clientes potenciales del subsistema y por tanto a los procesos que se necesitan informatizar.

### **Técnicas:**

**Entrevista:** Se realizaron múltiples entrevistas a especialistas que laboran en el Departamento de Economía con el fin de obtener conocimientos más específicos de cómo se gestiona la información relacionada con el control de los activos fijos tangibles en el Instituto de Suelos de Granma, quienes manejan la información y quiénes son los encargados de dicho proceso.

## **Capítulo 1. Fundamentos para la gestión y control de los activos fijos tangibles**

### **1.1 Introducción**

En el presente capítulo se hace un análisis del objeto de estudio y el campo de acción de la investigación para la selección de los procesos a informatizar. Se abordan las tendencias y tecnologías actuales en cuanto a lenguajes de programación Web, servicios Web, diseño de interfaz, sistemas gestores de bases de datos y metodologías para el desarrollo de sistemas informáticos; haciendo énfasis en las que se emplean en la presente investigación.

### **1.2 Proceso de gestión de activos fijos tangibles**

Cuba se encuentra en un proceso de institucionalización, se han venido estableciendo una serie de disposiciones legales que constituyen todo un sistema normativo, que es toda una filosofía de gestión de la organización que se concreta en la misión, visión, valores y de una necesaria comprensión del conjunto de códigos, leyes, resoluciones, reglamentos en el contexto de estas exigencias.

El 1ro de octubre del 2009, la Contraloría General de la República de Cuba hace llegar a las entidades presupuestadas, la guía de autocontrol del sistema de control interno para que cada entidad autoevalúe la calidad de los procesos que realiza, en dicha guía se establecen la serie de aspectos a verificar separadas por grupos o actividades pudiéndose apreciar que desde el aspecto 201 hasta el 221 corresponden al control de los activos fijos tangibles (de Armas Phillips y Machado Barrios, 2010).

#### **1.2.1 Gestión de activos fijos tangibles en el sector empresarial estatal cubano**

En las empresas cubanas, los Activos Fijos presentan un grupo de medidas para su preservación teniendo en cuenta que estos deben estar asegurados por un Modelo de control de los mismos. El jefe de cada una de las áreas de cada empresa presenta un acta de responsabilidad material siendo el máximo responsable, en caso de ser varios los responsables se le adjunta un anexo llamado "Responsabilidad Material Colectiva" donde aparezca el nombre y apellidos, cargo que ocupa y firma de cada responsable. Anualmente, a



principio de año, se confecciona un plan de chequeo del 10 %, en caso de un cambio del responsable se le realiza al área un chequeo del 100 % procediendo inmediatamente al cambio del acta de responsabilidad material. En caso de algún movimiento interno dentro de cualquier área de cualquier empresa ya sea bajas, altas o traslados se ajusta la hoja de control y se le comunica al responsable para que sea de su conocimiento. En caso de que existan sobrantes y faltantes ya sea por la realización de un inventario físico o por otra circunstancia se inicia el expediente para la investigación de las causas de dicho sobrante o faltante considerándose un gasto no deducible, las pérdidas no cubiertas por el seguro motivadas por actitud negligente comprobada y determinada la responsabilidad material al responsable se le aplica el decreto ley 249/07 tomando en cuenta que al trabajador se le cobra en moneda nacional descontándosele mediante la nómina.

### **1.2.2 Gestión de activos fijos tangibles en el Instituto de Suelos de Granma**

El Instituto de Suelos de Granma es el encargado de promover y garantizar un uso, manejo y conservación de los suelos en el territorio, está formado por áreas o grupos de trabajo, estructurados cada uno de acuerdo a la responsabilidad en el cumplimiento del objeto social de la empresa. De estos grupos es el de economía y servicios el encargado de la gestión y control de los activos fijos tangibles, tarea de gran importancia para mantener la calidad de los procesos fundamentales de la institución.

Dicha gestión se realiza, como se explicó en la introducción, mediante el uso del tabulador electrónico *Microsoft Excel*. Las hojas de cálculo de Excel, para los usuarios no especialistas en informática, no garantizan correctamente la seguridad de los datos, permiten la repetición de datos innecesariamente, permiten la introducción de datos erróneos, complejizan y retrasan la obtención de reportes necesarios para la toma de decisiones de los responsables de esta tarea.

### 1.3 Sistemas informáticos para la gestión de activos fijos tangibles

#### ASSETS

Dentro de los Sistemas de Gestión para las aplicaciones económico-contable, se encuentra el **ASSETS**, sistema para la Gestión Integral estándar y parametrizado, es un sistema italiano que controla la actividad económica, contable y de Recursos Humanos, perteneciente al Ing. Informático Marco de Lucas con dos versiones; la primera sobre *Microsoft Access* y la segunda versión conocida como “Nueva Solución” usa el modelo cliente-servidor, como gestor de base de datos *Microsoft SQL Server 2000* y *Crystal Reports 8.5* para la generación de reportes de salidas (ASSETS, 2006; Cano Hidalgo, 2006).

Permite el control de los procesos de Compras, Ventas, Producción, Taller, Inventario, Finanzas, Contabilidad, Presupuesto, Activos Fijos, Útiles y Herramientas y Recursos Humanos. Como Sistema Integral todos sus módulos trabajan en estrecha relación, generando, automáticamente, al Módulo de Contabilidad los Comprobantes de Operaciones por cada una de las transacciones efectuadas, esto permite que se pueda trabajar bajo el principio de Contabilidad al Día.

Es un sistema flexible, amigable, con ayuda en línea que puede ser instalado en una microcomputadora o sobre varias, funcionando en ambiente multiusuario incluidas estaciones remotas. Asimismo, proporciona opciones de seguridad que le permiten limitar el acceso a los diferentes procesos del sistema de acuerdo con el perfil de cada usuario, garantizando que sus reportes tengan la forma y el contenido que el usuario les defina.

#### Beneficios y ventajas

- ASSETS NS es una aplicación cliente-servidor programada en *Visual Basic 6.0* y *Microsoft SQL Server 2000*, utilizando adicionalmente *Crystal Reports 7.0* para la generación de reportes de salidas
- Genera, automáticamente, los asientos de diario a la contabilidad por cada una de las transacciones contempladas en el sistema
- Con respecto a versiones anteriores, ASSETS NS garantiza un mejor rendimiento de la aplicación dada su filosofía cliente-servidor, que

proporciona mayor rapidez y eficiencia en la operación de los diferentes procesos y en la obtención de resultados

- Al estar en plataforma SQL, garantiza mayor seguridad y consistencia en los datos, se obliga que sea ilimitado el número de usuarios conectados y hace posible la utilización de servidores remotos
- Todos los procesos están implementados con inicio y final de transacciones, lo que garantiza la integridad de la base de datos ante fallos de corriente, cambios de voltaje o cualquier otra eventualidad que provoque una falla en la operación del sistema
- Al ser ASSETS NS una aplicación cliente-servidor realiza una manipulación mínima de datos en el sitio del cliente, todos los procesos se ejecutan en el servidor viajando al usuario las respuestas a cada uno de los procesos, lo que garantiza por una parte, que la carga en la red no sea significativa, permitiendo tener mayor número de usuarios conectados y, por otra, abarata los costos, dado que el hardware en el puesto del cliente no tiene grandes exigencias en cuanto a recursos

A pesar de sus ventajas el ASSETS es un software propietario que no posibilita el acceso o publicación de un gran número de informaciones útiles para el personal no especializado en Recursos Humanos, es un sistema orientado estrictamente a los locales de trabajo de los especialistas en la materia, careciendo por lo tanto de una interfaz que permita un flujo de informaciones haciendo nula la comunicación externa.

### **CEDRUX**

El Cedrux: Sistema Integral de Gestión, es una aplicación informática que se construye especialmente para el control de los recursos empresariales de las entidades, conocidos internacionalmente como sistemas ERP por sus siglas en inglés (*Enterprise Resource Planning*) Cedrux es un paquete de soluciones integrales de gestión para las entidades presupuestadas y empresariales.

Cedrux es un sistema integrado de gestión que se desarrolla en la Universidad de las Ciencias Informáticas (La Habana). Este sistema posee tres características que los identifican:

- Su sistema integral.

- Su modularidad.
- Su adaptabilidad.

## **VERSAT**

VERSAT-Sarasola conocido como “un paquete integrado para la gestión económica financiera que permite enviar información eficaz, de forma inmediata, desde lugares apartados, a la vez que ofrece mayor organización, control y disciplina en cada gestión”.

Se distingue por ser el primer sistema de contabilidad cubano certificado, según las nuevas normativas establecidas por los Ministerios de Finanzas y Precios y de la Informática y las Comunicaciones, para este tipo de Software.

VERSAT es un sistema económico integrado constituido por 10 módulos o subsistemas que incluyen configuración y seguridad, contabilidad general y de gastos, costos y procesos, finanzas y caja. Además, en el proceso intervienen activos fijos, planificación y presupuestos, control de inventarios, pago de salario (nómina), facturación y generador de reportes.

### **Características del VERSAT SARASOLA**

- Herramienta para la planificación económica, el control y el análisis de gestión.
- Diseñado para su empleo en cualquier tipo de entidad empresarial o presupuestada.
- Permite llevar el control y registro contable individual de todos los hechos económicos que se originan en las estructuras internas de las entidades, así como exponer el estado financiero y toda la información económica y contable en este universo.
- Se estructura en un grupo de subsistemas en los cuales se procesan y contabilizan los documentos primarios, donde se anotan los movimientos, los recursos materiales, laborales y financieros que se utilizan en una entidad.
- Se logra establecer un proceso de interacción usuario-sistema.
- Rapidez y fiabilidad, a partir de la configuración del proceso de contabilización de los documentos primarios y de las propias posibilidades de trabajo contenidas en cada subsistema.

El VERSAT-Sarasola, a pesar de ser una poderosa herramienta, versátil, completa y altamente configurable, es también software propietario, y el costo de su licencia de instalación y uso oscila alrededor de los 10000 pesos y la licencia de su código sobrepasa los 20000 pesos, difíciles de acceder para las empresas presupuestadas.

### **Sistema de inventario de la Facultad de Informática de la Universidad de Granma**

Es un sistema web, desarrollado por Marco Vinicio Cuyo Sigcha para el control de los medios básicos de la facultad de Informática de la Universidad de Granma, para su construcción se emplearon tecnologías libres lo que lo hace bastante económico, es un sistema modular, debido a que su gestión es solo para medios básicos es bastante rápido, es flexible y fácilmente adaptable a otras facultades y universidades.

El mayor inconveniente es que se desarrolló estrictamente basándose en las características y estructuras propias de las universidades, que las diferencian de otras empresas por lo que su uso se limita solo a este tipo de instituciones.

## **1.4 Herramientas, tecnologías y metodologías para el desarrollo de sistemas informáticos**

### **1.4.1 Aplicaciones web**

Una Aplicación Web es un conjunto de páginas Web enlazadas que visualizan diferentes partes de la información que se quiere mostrar a través de ella. Constituye una de las mejores herramientas para divulgar, gestionar y compartir la información por lo que trae consigo un aumento de la eficiencia en cuanto a la manipulación de gran cantidad de información. Permiten modificar la lógica del negocio a través de un navegador (Cinca, 2008; Hernán Ruiz, 2006).

### **Surgimiento de las Aplicaciones Web**

La *World Wide Web* (WWW): o simplemente "la Web", puede definirse básicamente como tres cosas: hipertexto, que es un sistema de enlaces que permite saltar de unos lugares a otros; multimedia, que hace referencia al tipo de contenidos que puede manejar (texto, gráficos, vídeo, sonido y otros) e Internet, las base sobre las que se transmite la información (Furgeta, 2002).

La WWW permite visualizar en la pantalla del usuario páginas con información alojadas en computadoras remotas (llamadas genéricamente sitios). En la actualidad, existen dos tipos de sitios Web: los que se comportan como magazines, donde solo se puede leer la información que en él esté escrita; y los que se comportan a manera de software, donde se ejecutan una serie de tareas específicas. Estos últimos son las llamadas aplicaciones Web (Cinca, 2008; Pressman, 1996; Sanz, 2002).

La arquitectura de un sitio Web es simple. Contiene como componentes principales: el Servidor Web, una Red y un Navegador o cliente. La arquitectura de una aplicación Web además incluye la aplicación en el Servidor, que es la que permite al sistema manejar lógica de negocio y tener un estado (Bellman, 2007).

Con el uso de las aplicaciones Web en cualquier entidad se logra, entre otros beneficios, una alta eficiencia que se alcanza en el área que haga uso de los mismos. Debido a una serie de características que ellos poseen, se mejoran los mecanismos de intercambio de información en la entidad, disminuyendo en un alto grado cualquier dificultad que exista para difundir la misma (Villa, 2006).

Entre las principales comodidades que una aplicación Web pueden traer se encuentran las siguientes:

- **Compatibilidad multiplataforma.** Las aplicaciones Web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software descargables. Varias tecnologías incluyendo Java, Flash, ASP y Ajax permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales (Colombo, Demetrescu, Finocchi, y Laura, 2003).
- **Actualización.** Las aplicaciones basadas en Web están siempre actualizadas con el último lanzamiento sin requerir que el usuario tome acciones pro-activas, y sin necesitar llamar la atención del usuario o interferir con sus hábitos de trabajo con la esperanza de que va a iniciar nuevas descargas y procedimientos de instalación (algunas veces imposible cuando usted está trabajando dentro de grandes organizaciones) (Oliver, 2000).

- **Inmediatez de acceso.** Las aplicaciones basadas en Web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online y estas están listas para trabajar sin importar cuál es su configuración o su hardware.
- **Facilidad de prueba.** Finalmente no habrá más obstáculos para permitir pruebas sencillas y efectivas de herramientas y aplicaciones antes de cargar su tarjeta de crédito. Actualmente, especialmente cuando hablamos de software costoso, hay todavía una gran cantidad de funcionalidades y pequeños detalles que no pueden ser totalmente probados descubiertos antes de comprometer dinero en alguna compra total.
- **Menos requerimientos de memoria.** Las aplicaciones basadas en Web tienen muchas más razonables demandas de memoria RAM de parte del usuario final que los programas instalados localmente. Al residir y correr en los servidores del proveedor, a esas aplicaciones basadas en Web usa en muchos casos la memoria de las computadoras que ellos corren, dejando más espacio para correr múltiples aplicaciones del mismo tiempo sin incurrir en frustrantes deterioros en el rendimiento.
- **Menos Bugs.** Las aplicaciones basadas en Web, deberían ser menos propensas a colgarse y crear problemas técnicos debido a software o conflictos de hardware con otras aplicaciones existentes, protocolos o software personal interno. Con aplicaciones basadas en Web, todos utilizan la misma versión, y todos los *bugs* pueden ser corregidos tan pronto como son descubiertos. Esta es la razón por la cual las aplicaciones basadas en Web deberían tener mucho menos bugs que el software de escritorio descargable tradicional (Zelkovitz, Shaw, y Gannon, 2009).
- **Precio.** Las aplicaciones basadas en Web no requieren la infraestructura de distribución, soporte técnico y marketing requerido por el software descargable tradicional. Esto permite que las aplicaciones online cuesten una fracción de sus contrapartes descargables si no totalmente gratuitas, mientras que ofrecen componentes adicionales y servicios Premium como una opción.

- **Múltiples usuarios concurrentes.** Las aplicaciones basadas en Web puedan realmente ser utilizada por múltiples usuarios al mismo tiempo. No hay más necesidad de compartir pantallas o enviar instantáneas cuando múltiples usuarios pueden ver e incluso editar el mismo documento de manera conjunta. Las compañías de conferencia Web y colaboración online están involucradas algunas transformaciones claves y los usuarios necesitan explorar que significa realmente trabajar efectivamente y como editar documentos juntos (Brusilovsky, Gilles, Frasson, y VanLehn, 2007).
- **Desarrollar aplicaciones en el lenguaje que usted quiera.** Una vez que las aplicaciones han sido separadas de computadoras locales y sistemas operativos específicos esos pueden también ser escritas en prácticamente cualquier lenguaje de programación. Ya que las aplicaciones Web son esencialmente una colección de programas más que un simple programa, ellas podrían ser escritas en cualquier lenguaje de programación que esté de aplicación Web. Mientras que para software escritorio usted está limitado a usar el mismo lenguaje que el sistema operativo subyacente este no es el caso cuando la aplicación de software (Brusilovsky, y otros., 2007).

Usando la Web, se tiene acceso a millones de páginas de información, la exploración se realiza por medio de un software especial denominado *Browser* o Explorador. Las aplicaciones Web son populares debido a la practicidad del navegador Web como cliente ligero (Conallen, 2006; Prentzas, Hatzilygeroudis, y Garofalakis, 2002).

### **Internet**

Internet (*INTERconnected NETWORK*) es una red de redes ordenadores de todo tipo que se comunican mediante un lenguaje común: el conocido como protocolo TCP/IP. Esa primera característica de la heterogeneidad de los equipos conectados es clave para entender el funcionamiento de todos los servicios de la red y para comprender la necesidad de la portabilidad en cualquier desarrollo que se quiera hacer en el ámbito de Internet. A nivel de programadores de aplicaciones web.

El desarrollo de Internet no solo se traduce en beneficio para los usuarios, sino que también lo es para los organismos, las instituciones, las empresas etc.



Internet se ha convertido en uno de los medios de comunicación más extendido en toda la historia de la humanidad. Está bien llamada "Red de redes", permite establecer la cooperación y colaboración entre gran número de comunidades y grupos de interés por temas específicos, distribuidos por todo el planeta (Lynch, 2003).

### **Intranet**

Es una infraestructura basada en los estándares y tecnologías de Internet que soporta el compartir información dentro de un grupo bien definido y limitado. Las Intranets son Internets internas basadas en particular en el sistema WWW, pero adaptándolo a los límites físicos y a las características particulares de cada organización (Herrera, Sandoval, y Quijada, 2009).

Una Intranet es una alternativa para las empresas que necesiten compartir la información que se genera en ella. Es utilizar y distribuir la información dentro de la empresa pero a la manera de Internet. Es una red de comunicaciones dentro de una empresa, que permite integrar una serie importante de recursos, pudiendo así obtener la máxima eficiencia.

### **El modelo cliente/servidor**

Todos los servicios que ofrece Internet, y por supuesto entre ellos el servicio web se basan en la denominada relación cliente /servidor. El comprender bien esta relación es esencial para entender el funcionamiento posterior de lenguajes como PHP. En Internet se puede encontrar dos tipos de equipos conectados.

- **Servidores:** ordenadores que ofrecen su servicio al resto de equipos conectados. Suelen tener presencia estable en la red, lo que se concreta en tener asignadas direcciones IP permanentes. En ellos es donde están alojadas por ejemplo, las páginas web.
- **Cliente:** equipos que los usuarios individuales utilizan para conectarse a la red y solicitar servicios a los servidores. Durante el tiempo de conexión tiene presencia física en la red. Normalmente los proveedores de acceso a Internet asignan a estos equipos una dirección IP durante su conexión, pero esa dirección es variable es decir, cambia de unas conexiones a otras (IP dinámicas).

Los conceptos de cliente y servidor se suelen utilizar con dos significados diferentes, en referencia al hardware el sentido es el indicado anteriormente, el servidor hace referencia al equipo remoto al que se realiza la conexión y el cliente sería el equipo local utilizado para efectuar dicha conexión. Pero también se utilizan esos conceptos en referencia al software.

- **Programa servidor:** es el programa que debe estar ejecutándose en equipo servidor para que este pueda ofrecer su servicio. Un documento HTML sin más almacenado en el equipo remoto no basta para que sea accesible como página web por el resto de usuarios de Internet. En ese equipo debe estar ejecutándose una aplicación servidor web. Uno de los programas servidores web más conocido y utilizado es Apache, programa que también pertenece a la corriente *open source*. Existen otros servidores web como el personal *web Server* (PWS) o el *Internet Information Server* (IIS). disponibles en los equipos de *Windows*. En el caso de otros servicios como el correo electrónico o la transferencia de ficheros se necesitarían igualmente los correspondientes programas en el servidor.
- **Programa cliente:** es en este caso el software necesario en el equipo cliente para tener acceso al correspondiente servicio. Así por ejemplo, los navegadores como el Internet Explore o Mozilla son ejemplos de clientes web; un programa como *Outlook* es un ejemplo de cliente de correo electrónico y programas WS\_FTP o CuteFTP son ejemplos FTP. (Cobo, Gómez, Pérez, y Rocha, 2005).

### 1.4.2 Lenguajes y librerías para el desarrollo de aplicaciones web

#### CSS

Las hojas de estilo en cascada, del inglés, *Cascade StyleSheets* (CSS) es un lenguaje de hojas de estilos creado para controlar la presentación de los documentos electrónicos definidos con HTML y XHTML. Es la mejor forma de separar contenido y presentación; es imprescindible para crear aplicaciones web complejas. Separar contenido y presentación, brinda numerosas ventajas, ya que obliga a crear documentos HTML/ XHTML bien definidos, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y

permite visualizar el mismo documento en infinidad de dispositivos diferentes. Este lenguaje se utiliza para definir el aspecto de todos los contenidos, el formato de tablas, la separación, el color, tamaño y tipo de letra de titulares y/o textos, la tabulación con la que se muestran los elementos de una lista o menú (Hernández y Greguas, 2010).

### ***Hypertext Markup Language (HTML)***

HTML es el lenguaje de la Web, estas siglas significan (Lenguaje de Marcas de Hipertextos), en donde hipertexto hace referencia a la capacidad del lenguaje para la utilización de los conocidos hipervínculos. Se basa en el uso de etiquetas o marcas, gracias a ellas es posible definir el formato del texto, los distintos elementos que conforman la página, sus propiedades y disposición. Este lenguaje es interpretado por los navegadores, procesado pasados y convertidos en una página Web (Málaga, 2008).

Una página Web o documento HTML no es otra cosa que un archivo de texto que se puede crear con cualquier editor; desde el bloc de notas Windows hasta cualquiera de DOS o LINUX.

Fue creado en 1986 por el físico nuclear Tim *Berners-Lee*; el cual tomo dos herramientas preexistentes: El concepto de Hipertexto (Conocido también como link o ancla) el cual permite conectar dos elementos entre si y el Lenguaje Estándar de Marcación General (SGML) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto algún error de sintaxis que se presente éste no lo detectará y se visualizara en la forma como éste lo entienda (Hernán Ruiz, 2006).

### **JavaScript**

JavaScript es un lenguaje de scripts desarrollado por Netscape para incrementar las funcionalidades del lenguaje HTML. Se utiliza embebido en el código HTML, entre las tags `<script>` y `</script>`. Sus características más importantes son:

Es un lenguaje interpretado basado en guiones, es decir, no requiere compilación. El navegador del usuario se encarga de interpretar las sentencias

Java Script contenidas en una página HTML y ejecutarlas adecuadamente (Colombo, y otros., 2003).

Java Script es un lenguaje orientado a objetos. El modelo de objetos de Java Script está reducido y simplificado, pero incluye los elementos necesarios para que los Scripts puedan acceder a la información de una página y puedan actuar sobre la interfaz del navegador.

Cuando un usuario selecciona un enlace o mueve el puntero sobre una imagen se produce un evento. Mediante JavaScript se pueden desarrollar Scripts que ejecuten acciones en respuesta a estos eventos. En este caso actúa como un lenguaje orientado a eventos (Málaga, 2008).

Es además un lenguaje de alto nivel multiplataforma que admite programación estructurada y no necesita ningún kit o entorno de desarrollo (Frentzen y Henry, 1999).

### **PHP**

El PHP (acrónimo de PHP: *Hypertext Preprocessor*), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. El PHP inicio como una modificación a Perl escrita por Rasmus Lerdorf a finales de 1994. Su primer uso fue el de mantener un control sobre quien visitaba su curriculum en su web. (Gilmore, 2006)

Comparado con ASP, la principal ventaja en PHP es su carácter multiplataforma.

Por otro lado los programas con ASP resultan más lentos y pesados, y también menos estables en los entornos *Microsoft* la ventaja de ASP es que los servidores web de *Microsoft* soportan directamente ASP sin necesidad de ninguna instalación adicional.

Señalar también la existencia de herramientas que permiten convertir programas en ASP al lenguaje PHP, una de las más conocidas es *asp2php*.

Comparando el lenguaje PHP con el lenguaje Perl, utilizado habitualmente en la programación CGI (*Common Gateway Interface*), puede decirse que PHP fue diseñado para desarrollo de scripts orientados a web, mientras que Perl fue diseñado para hacer muchas más cosas y debido a esto, se hace muy

complicado. La sintaxis de PHP es menos confusa y más estricta, pero perder flexibilidad.

En comparación con *ColdFusion*, PHP es más rápido y eficiente para tareas complejas de programación, además PHP resulta más estable y usa una menor cantidad de recursos. Por el contrario. *ColdFusion* posee un mejor gestor de errores, un buen motor de búsqueda, abstracciones de bases de datos y un gran número de funcionalidades para el procesamiento de fechas finalmente, *ColdFusion* no está disponible para todas las plataformas.

En definitiva, PHP es uno de los lenguajes muy utilizado en el desarrollo de aplicaciones web y viene experimentando un constante crecimiento en su nivel de utilización en Internet (Cobo, y otros., 2005).

### **VBScript**

VBScript es, al igual que JavaScript, un lenguaje basado en guiones que permite integrar programas directamente en el código HTML. Admite un doble uso, por un lado como lenguaje del lado del cliente, pero también como lenguaje del lado del servidor para la generación de páginas *Active Server Pages* (ASP).

Es un lenguaje desarrollado por *Microsoft* tomando como referente de sintaxis el VBA, Visual Basic para aplicaciones. Por supuesto, no ofrece todas las funcionalidades de un entorno de desarrollo visual como Visual Basic, pero si se presenta como una herramienta poderosa y de fácil uso para generar páginas Web interactivas (ASKALL ORG, 2007).

### **Java**

Java es un lenguaje de programación clásico en cuanto a que requiere un proceso de compilación. El código compilado puede ser integrado en la página Web para ser ejecutado por el cliente.

Su nacimiento data de enero de 1996, tras el lanzamiento por la empresa creadora *Sun Microsystem* del JDK 1.0 (Java Development Kit). Es un entorno que puede obtenerse de forma gratuita e incorpora las herramientas básicas para la creación de una aplicación en Java (JDK, 2009).

Con Java se pueden crear dos tipos de programas:

- *Applets*: programas que se integran en las páginas Web y que, residiendo en el servidor, son ejecutados por el cliente. La ejecución necesita de la interpretación del código compilado por el software cliente (ASKALL ORG, 2007).
- Aplicaciones: Programas autónomos que se pueden ejecutar en cualquier equipo. En este último caso puede optarse por generar código compilado similar al de los *Applets* y que para su ejecución necesita de un intérprete o código compilado ejecutable directamente como en cualquier otro lenguaje de programación.

En el caso de *Applets*, el código fuente no se incrusta directamente en el documento HTML, si no al que se lo añade es un código binario resultado de la compilación, el denominado *Java Byte Code* (JBC). Esto permite proteger el código fuente, aunque hasta cierto punto, ya que las particularidades de este código compilado hacen que sea factible el proceso inverso, es decir, la de compilación, recuperar el código fuente a partir del compilado. En la propia Internet pueden encontrarse programas capaces de hacerlo. La razón de todo esto está en el hecho de que para conseguir la portabilidad de los programas el código compilado es un código que se encuentra entre el código fuente y un código objeto fuertemente dependiente de una plataforma. Es por ello que se suele decir de Java que es un lenguaje que combina la flexibilidad de los lenguajes interpretados y el poder de los compilados (JBC, 2009).

### 1.4.3 Servidores web

Un servidor Web es un programa que implementa el protocolo HTTP. Este protocolo está diseñado para transferir lo que se llama hipertextos, páginas Web o páginas HTML: textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonido (Valles, 2008).

Un servidor Web se encarga de mantenerse a la espera de peticiones HTTP llevada a cabo por un cliente HTTP o navegador. El navegador realiza una petición al servidor y este le responde con el contenido que el cliente solicita. Al solicitar una dirección de una página Web en un navegador, este realiza una petición HTTP al servidor de dicha dirección. El servidor responde al cliente

enviando el código HTML de la página. El cliente, una vez recibido el código, lo interpreta y lo muestra en pantalla. Entre los servidores Web más importantes se encuentra el Apache e *Internet Information Server* (IIS). Sobre ambos de amplia a continuación.

### ***Internet Information Server (IIS)***

*Internet Information Services* o IIS es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows. Originalmente era parte del *Option Pack* para *Windows NT*. Luego fue integrado en otros sistemas operativos de *Microsoft* destinados a ofrecer servicios, como *Windows 2000* o *Windows Server 2003*. *Windows XP Profesional* incluye una versión limitada de IIS. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS.

Antiguamente se denominaba PWS (*Personal Web Server*), y actualmente forma parte de la distribución estándar de *Windows*, de modo que no se necesita una licencia extra para instalarlo. Este servicio convierte a una PC en un servidor web para Internet o una *intranet*, es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente.

Los servicios de *Internet Information Services* proporcionan las herramientas y funciones necesarias para administrar de forma sencilla un servidor web seguro.

El servidor web se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas. Por ejemplo, *Microsoft* incluye los de *Active Server Pages* (ASP) y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP o Perl.

### **Apache**

Apache es el servidor web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa.

- Corre en una multitud de sistemas operativos, lo que lo hace prácticamente universal.

- Apache es una tecnología gratuita de código fuente abierto. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esto le da una transparencia a este software de manera que si se quiere ver lo que se está instalando como servidor, se puede saber, sin ningún secreto, sin ninguna puerta trasera.
- Apache es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que se instalen cuando se necesiten. Otra cosa importante es que cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un módulo para realizar una función determinada.
- Apache trabaja con gran cantidad de Perl, PHP y otros lenguajes de *script*. Perl destaca en el mundo del *script* y Apache utiliza su parte del pastel de Perl tanto con soporte CGI como con soporte *mod perl*. También trabaja con Java y páginas *jsp*. Teniendo todo el soporte que se necesita para tener páginas dinámicas.
- Apache permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.
- Tiene una alta configurabilidad en la creación y gestión de *logs*. Apache permite la creación de ficheros de log a medida del administrador, de este modo puedes tener un mayor control sobre lo que sucede en tu servidor.

#### 1.4.4 Sistemas de administración de bases de datos

Un Sistema de Administración de Bases de Datos o Sistema de Gestión de Bases de Datos (SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos, además de proporcionar herramientas para añadir, borrar modificar y analizar los datos. Los usuarios pueden acceder a la información usando herramientas específicas de interrogación y de generación de informes, o bien mediante aplicaciones al efecto.



Los SGBD también proporcionan métodos para mantener la integridad de los datos, para administrar el acceso de usuarios a los datos y recuperar la información si el sistema se corrompe. Permite presentar la información de la base de datos en variados formatos. La mayoría de los SGBD incluyen un generador de informes. También puede incluir un módulo gráfico que permita presentar la información con gráficos y tartas.

Hay muchos tipos de SGBD distintos según manejen los datos y muchos tamaños distintos según funcionen sobre ordenadores personales y con poca memoria a grandes sistemas que funcionan en *mainframes* con sistemas de almacenamiento especiales.

Generalmente se accede a los datos mediante lenguajes de interrogación, lenguajes de alto nivel que simplifican la tarea de construir las aplicaciones. También simplifican la interrogación y la presentación de la información. Un SGBD permite controlar el acceso a los datos, asegurar su integridad, gestionar el acceso concurrente a ellos, recuperar los datos tras un fallo del sistema y hacer copias de seguridad. Las Bases de Datos y los sistemas para su gestión son esenciales para cualquier área de negocio, y deben ser gestionados con esmero.

### **SQL Server**

*Microsoft SQL Server*, propietario de *Microsoft*, pertenece a la familia de los sistemas de administración de base de datos, operando en una arquitectura cliente/servidor de gran rendimiento. Su desarrollo fue orientado para hacer posible manejar grandes volúmenes de información, y un elevado número de transacciones. SQL Server es una aplicación completa que realiza toda la gestión relacionada con los datos. El servidor sólo tiene que enviarle una cadena de caracteres (la sentencia SQL) y esperar a que le devuelvan los datos (SQL, 2007).

SQL Server permite la creación de procedimientos almacenados, los cuales consisten en instrucciones SQL que se almacenan dentro de una base de datos de SQL Server, realizados en lenguaje SQL, se trata de procedimientos que se guardan semicompilados en el servidor y que pueden ser invocados

desde el cliente. Se ejecutan más rápido que instrucciones SQL independientes.

SQL Server puede manejar perfectamente bases de datos de *TeraBytes* con millones de registros y funciona sin problemas con miles de conexiones simultáneas a los datos, sólo depende de la potencia del hardware del equipo en el que esté instalado.

### **PostgreSQL**

Surge del paquete de Postgres desarrollado en la universidad de Berkeley en California. Con más de una década de desarrollo como respaldo, PostgreSQL es uno de los sistemas de Base de Datos libre avanzados que está disponible en el mundo hoy.

PostgreSQL tiene transacciones, integridad referencial, vistas, y multitud de funcionalidades, pero es lento y pesado que MySQL, aunque en las últimas versiones del mismo esto ha mejorado mucho (I.M. Lockhart, 2009).

En PostgreSQL se utilizan las vistas como tablas virtuales, como una tabla que no existe físicamente en la Base de Datos, pero aparece al usuario como si existiera. Cuando se habla de una tabla base, se refiere a que realmente hay un equivalente almacenado para cada fila en la tabla en algún sitio del almacenamiento físico.

Uno de los principios del modelo relacional es que los atributos de una relación son atómicos. PostgreSQL no contiene esta restricción; los atributos pueden contener subvalores a los que puede accederse desde el lenguaje de consulta (Lockhart, 2009).

### **MySQL**

MySQL es un servidor de base de datos. Lo que durante un tiempo se consideró como un sencillo juguete para su uso en sitios Web, se ha convertido en la actualidad en una solución viable y de misión crítica para la administración de datos. Antes, MySQL se consideraba como la opción ideal para sitios Web; sin embargo, incorpora muchas de las funciones necesarias para otros entornos y conserva su gran velocidad. MySQL supera desde hace tiempo a muchas soluciones comerciales en velocidad y dispone de un sistema

de permisos elegante y potente, y ahora, además, incluye el motor de almacenamiento InnoDB compatible con ACID (Atomicidad, Consistencia, Independencia y Durabilidad).

MySQL es rápido, dispone de funciones de volcado online e incorpora una gran cantidad de funciones nuevas. Son pocas las razones para desechar MySQL como solución de base de datos. MySQL AB, la compañía responsable del desarrollo de MySQL, dispone de un sistema de asistencia eficiente y a un precio razonable, y, como ocurre con la mayor parte de las comunidades de código abierto, encontrará una gran cantidad de ayuda en la Web. (Gilfillan, 2004)

### **1.5 Metodologías de desarrollo de software**

Una metodología de desarrollo de software se refiere al entorno que se usa para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información. Una gran variedad de metodologías se han desarrollado a lo largo de los años, cada una de ellas con sus fortalezas y debilidades. Una determinada metodología no es necesariamente aplicable a todo tipo de proyectos, más bien cada tipo de proyecto tiene una metodología a la que se adapta mejor.

#### **1.5.1 Proceso Unificado de desarrollo de software (RUP)**

La metodología Proceso Unificado de Desarrollo de Software (RUP) se basa en la orientación a objetos, el desarrollo iterativo y el modelamiento visual usando el Lenguaje Unificado de Modelación (UML) para la ingeniería de sistemas y de software, lo que permite incorporar al proceso de desarrollo de software un mejor control de los requerimientos y cambios. Esta metodología proporciona al equipo del proyecto procedimientos y herramientas promoviendo las siguientes prácticas (Jacobson, Booch, y Rumbaugh, 2000):

- Gestión de los Requisitos.
- Desarrollos iterativos.
- Uso de arquitecturas basadas en componentes.
- Desarrollo Visual del Software con UML.
- Verificación continua de la calidad del software.
- Gestión de los Cambios.

Según la metodología RUP el ciclo de vida de un proyecto se divide en las siguientes fases:

- Fase de concepción.
- Fase de elaboración.
- Fase de construcción.
- Fase de transición (Bowden, 2007).

A pesar de ser una metodología desarrollada directamente para el trabajo con clases y objetos brinda amplias posibilidades para el manejo eficiente del tiempo de diseño e implementación utilizando las extensiones para aplicaciones Web. (Furgeta, 2002).

### **1.5.2 Programación Extrema (XP)**

*Extreme Programming* más conocida por sus siglas (XP), es una metodología ágil de desarrollo de software que se basa en la simplicidad, la comunicación y la realimentación o reutilización del código desarrollado. Surgió como respuesta y posible solución a los problemas derivados del cambio en los requerimientos, se plantea como una metodología a emplear en proyectos de riesgo y aumenta la productividad (Ávila A & Pérez B, 2009)

#### **Características de XP**

- ❖ Se basa en:
  - Pruebas unitarias: son las pruebas realizadas a los principales procesos, de tal manera que adelantándose en algo hacia el futuro, se puedan hacer pruebas de las fallas que pudieran ocurrir. Es como adelantarse para luego obtener los posibles errores.
  - Refabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
  - Programación en pares: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.
- ❖ Propone:

- Empezar en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- No introduce funcionalidades antes que sean necesarias.
- El cliente o el usuario se convierten en miembro del equipo (Ávila A y Pérez B, 2009).

**¿Por qué utilizar XP?** El equipo de desarrollo para darle solución al problema planteado en la investigación es de dos personas; además, el espacio de tiempo para solucionarlo es corto y no sería conveniente generar exceso de documentación para un proyecto de pequeña dimensión. El cliente está en habitual intercambio con el equipo de desarrollo y como el proyecto aun es joven está constantemente generando nuevos procesos de trabajo; por lo que la utilización de una metodología adaptable al cambio es lo más conveniente. Por esto y todo lo anterior planteado, XP constituye la metodología adecuada para guiar el proceso de desarrollo de la solución propuesta.

### **1.6 Justificación de las herramientas seleccionadas**

La selección de las tecnologías a utilizar se realizó teniendo en cuenta las prestaciones de cada una de ellas y la plataforma de desarrollo en que se desea construir la aplicación, a modo de mantener la mayor integración posible entre las mismas. Además se tuvieron en cuenta las líneas y pautas del proyecto, así como las exigencias del cliente. El desarrollo del sistema propuesto será guiado por la metodología de desarrollo Programación Extrema, más conocida como XP; es la metodología de desarrollo de software más exitosa en la actualidad, utilizada para proyectos de corto plazo y corto equipo. Se implementará haciendo uso de los lenguajes de programación: HTML que es un lenguaje de marcado predominante para la creación y publicación de páginas web al que se le pueden añadir características y funcionalidades mediante las Hojas de Estilo (CSS) y JavaScript dando como resultado páginas web rápidas y sencillas; PHP que es un lenguaje del lado del servidor diseñado específicamente para el desarrollo de páginas web dinámicas que puede ser incluido con facilidad en el código HTML, incorpora una gran cantidad de funciones para realizar útiles tareas relacionadas con la web, es libre y

multiplataforma. El servidor web que se utilizará será el Apache que ofrece un amplio soporte de PHP, es una tecnología gratuita de código fuente abierto, es multiplataforma, extensible y está considerado como uno de los más utilizados actualmente. Como gestor de bases de datos se utilizará MySQL debido a que teniendo en cuenta que forma un buen equipo con servidor web Apache para servir páginas Web con contenido dinámico. Para crear y editar las páginas Web del sistema se eligió la herramienta de desarrollo *Dreamweaver CS5* ya que con la misma se podrá desarrollar cualquier sistema web personal con características de sistema profesional y utilizar casi todos los recursos de la web, así como realizar aplicaciones que se ejecuten en el servidor y vinculaciones dinámicas de datos; además de contar con un soporte para aplicaciones PHP.

### **1.7 Conclusiones del capítulo**

Al finalizar el capítulo se ha expuesto todo lo referente a la base teórica, que fundamenta la presente investigación. Se tiene una valoración sobre el porqué de las elecciones referentes al lenguaje de programación a usar, el sistema gestor de bases de datos, la herramienta de desarrollo, el servidor Web y la metodología de ingeniería del software.

Luego de un análisis de algunas propuestas se escogió PHP versión 5.3.8 como lenguaje de programación, integrado con el servidor Web Apache versión 2.2.21 y con el servidor de bases de datos MySQL versión 5.5.16. La herramienta de desarrollo *Dreamweaver CS5* fue elegida para crear y editar las páginas Web del Sistema y para guiar el desarrollo la metodología *Rational Unified Process* (RUP) utilizándose en el modelamiento visual *Unified Modeling Language* (UML).

## Capítulo 2 Descripción, construcción y valoración de la propuesta de solución

### 2.1 Introducción

Luego de haber realizado un estudio del arte y de haber seleccionado las herramientas y la metodología que se utilizarán, se está en condiciones de realizar una descripción de las características principales del sistema a desarrollar, de acuerdo al problema por el cual fue concebido. Se especifican las funcionalidades que se desean informatizar en la búsqueda de satisfacer las necesidades de los clientes.

Este capítulo está enmarcado en las fases de exploración, planificación, iteraciones, producción, mantenimiento y muerte del proyecto, definidas en la metodología escogida. Además, se describe la construcción de la propuesta con el desarrollo de las historias de usuario a través de iteraciones, se definen tareas y principios de diseño para su implementación. Se exponen pruebas de aceptación para garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable.

### 2.2 Personas relacionadas con el sistema

Se define como persona relacionada con el sistema toda aquella que interactúa e intercambia con este y obtiene resultados de los procesos desarrollados en la aplicación. Además los sujetos que interactúan con la misma sin poder hacer uso de las secciones privilegiadas del sistema. En la siguiente tabla se detalla una breve descripción sobre las actividades que puede realizar una persona con la aplicación.

**Tabla 2.1** Personas relacionadas con el sistema

Persona	Justificación
Administrador del sistema	Es la persona encargada de gestionar usuarios, áreas y locales de trabajo.
Auxiliar económico	Es el encargado de gestionar datos de trabajadores, activos fijos tangibles y las operaciones que se realizan sobre estos.

Jefe de área	Es el encargado de asignar responsables a los activos fijos tangibles y a los locales de trabajo de su área.
Usuarios	Solo pueden visualizar información acerca de los activos fijos tangibles.

### 2.3 Fase de exploración

La metodología de desarrollo XP comienza con la fase de exploración, en esta fase se define el alcance del proyecto y al mismo tiempo el equipo de trabajo se familiariza con las herramientas y tecnologías que se utilizan en el proyecto, realizándose las historias de los usuarios que no son más que la forma de representar los requisitos del sistema a implementar. La duración de esta etapa puede variar en dependencia de la familiarización que tengan los desarrolladores con las herramientas.

#### 2.3.1 Historia de usuario

Las historias de usuario (HU) son la forma en que se especifican en XP los requisitos funcionales del sistema. Estas son generadas desde la perspectiva del cliente, aunque los desarrolladores pueden brindar también su ayuda en la identificación de las mismas. El contenido de estas debe ser concreto y sencillo. El nivel de detalle de las historias de usuarios debe de ser el mínimo posible que permita hacerse una idea de cuánto costará realizar la implementación del sistema. A continuación se muestran algunas historias de usuarios definidas en el proceso de desarrollo, se pueden consultar otras en el [Anexo 1](#).

**Tabla 2.2** Historia de usuario “Gestionar áreas”.

Historia de Usuario	
Número: 4	Usuario: Especialista económico
Nombre de historia: Gestionar área	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta



Puntos Estimados: 0.4	Iteración Asignada: 1
Programador responsable: Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
Descripción: Se podrá insertar, eliminar, modificar y visualizar el área.	
Observaciones:	

**Tabla 2.3** Historia de usuario “Gestionar activos fijos tangibles”.

Historia de Usuario	
Número: 6	Usuario: Especialista económico
Nombre de Historia: Gestionar activos fijos tangibles	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Alta
Puntos Estimados: 0.4	Iteración Asignada: 1
Programador Responsable: Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo.	
Descripción: El especialista económico podrá insertar, modificar, eliminar o visualizar activos fijos tangibles.	
Observaciones: Para el caso de insertar y modificar es necesario que existan áreas y locales registrados en el sistema.	

**Tabla 2.4** Historia de usuario “Gestionar control de servicios”.

Historia de Usuario	
Número: 7	Usuario: Especialista económico
Nombre de Historia: Gestionar control de servicios	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Alta
Puntos Estimados: 0.6	Iteración Asignada: 1

Programador Responsable: Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo.
Descripción: El especialista económico podrá insertar, modificar, eliminar o visualizar los servicios prestados a los activos fijos tangibles defectuosos.
Observaciones: Para gestionar los servicios prestados, deben existir medios básicos registrados en la base de datos.

**Tabla 2.5** Historia de usuario “Gestionar movimiento”.

Historia de Usuario	
Número: 8	Usuario: Especialista económico
Nombre de Historia: Gestionar movimiento	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Alta
Puntos Estimados: 0.6	Iteración Asignada: 2
Programador Responsable: Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
Descripción: El especialista económico podrá insertar, modificar, eliminar o visualizar los movimientos de activos fijos tangibles.	
Observaciones: Para realizar los respectivos movimientos tendrá que ser autorizado, aprobado y transportado por un trabajador.	

### 2.3.2 Valoración de sostenibilidad

La sociedad aspira a un mundo mejor en los aspectos económico, social y ambiental. El diseño e implementación de un sistema de gestión informático sustentable es un proceso de cambio progresivo en la calidad de vida del ser humano, que lo coloca como centro y sujeto primordial del desarrollo, por medio del crecimiento económico con equidad social y la transformación de los métodos de producción de los patrones de consumo y que se sustenta en el equilibrio ecológico y el soporte vital de la región (Concepción García, 2000).

Se conceptualiza la gestión de proyectos informáticos sostenibles como el proceso de planificación, organización, ejecución y control de acciones interdisciplinarias que permiten tomar las decisiones más adecuadas para un desarrollo informático sostenible en el entorno.

La gestión de sostenibilidad de un producto informático (PI) debe atender las dimensiones administrativas (administración de recursos), socio-humanistas (modo de vida, desarrollo de un grupo social, satisfacción de necesidades), ambientales (condiciones favorables o no a las personas o cosas, minimiza daños e impactos) y tecnológicas (uso de tecnología adecuada y asimilable con el usuario) del mismo. La valoración preventiva de alteraciones favorables o desfavorables en estas dimensiones contribuirá al desarrollo sostenible de un producto informático.

### **Dimensión Administrativa**

Para el desarrollo del “Sistema de control de activos fijos tangibles del Instituto de Suelos de Granma” se realizó una estimación de esfuerzos por historias de usuarios identificadas, permitiendo tener una medida real de la velocidad de progreso del proyecto y proporcionando una guía de tiempo razonable a la cual ajustarse.

El progreso del sistema no conlleva a un gasto de mano de obra, pues forma parte de un trabajo de tesis, el proceso no necesita equipamiento extra, ya que con el existente en la Universidad de Granma se puede desarrollar.

El lenguaje de programación a utilizar es PHP y *JavaScript*, como servidor de aplicaciones web *Apache* y el sistema gestor de base de datos *MySQL* los cuales son gratuitos, por lo que no se incurrirá en gastos de licencia o adquisición.

### **Dimensión Socio - Humanista**

En Cuba es muy importante el factor humano, por lo que a diario cada institución se esfuerza por brindar los trabajadores las comodidades indispensables para que ejerzan sus funciones dentro de la organización. Esto trae consigo un mayor provecho en las tareas que ellos realizan y mejoras en la calidad de vida.

El sistema no generará desempleo sino más bien facilitará en gran medida la realización de las actividades laborales de los usuarios, aumentando la confiabilidad y rapidez en la gestión de la información, además el software contribuirá a elevar el nivel profesional y cultural de todos los trabajadores ya que estarán interactuando con la tecnologías de avanzada en Cuba y el mundo en cuanto a medios de informática y comunicaciones.

Todo el trabajo con el software se realiza con un mínimo de esfuerzo. La aplicación no trae consigo más carga de trabajo, sino que la reduce en gran medida, evitando que la jornada laboral sea tediosa y provoque estrés en los trabajadores.

Por todo lo antes dicho se concluye que es sostenible el “Sistema de control de activos fijos tangibles del Instituto de Suelos de Granma”, de acuerdo a la dimensión socio-humanista.

### **Dimensión Ambiental**

La interfaz gráfica que brinda el sistema al usuario resulta agradable, presenta colores adecuados cuidando en todo momento el uso de los mismos, los cuales son familiares al usuario debido a que la mayoría de las aplicaciones con la que trabaja se encuentran en esta gama de colores y no resultan dañino a la vista lo que no representa riesgo psicológico alguno, además contiene otras interfaces con las mismas característica de las cuales puede elegir la que le resulte más agradable.

Se recomienda a los usuarios una correcta postura en las sillas, usar protectores de pantalla para cuidar la visión, así como poner el monitor a la altura de los ojos y frente a ellos, para evitar los problemas de visión, de columna y de cervical. Además se tuvo en cuenta las necesidades de los usuarios, ya que se utilizó un tamaño de letra adecuado para la correcta visualización del contenido. Con la implantación del software no se contribuye de modo alguno al deterioro gradual del medio ambiente sino todo lo contrario, implantándolo se da un paso más en el cuidado del mismo por lo que se considera que es sostenible teniendo en cuenta la dimensión medio ambiental.

### **Dimensión Tecnológica**

Referente a la dimensión tecnológica se puede decir que es necesaria la capacitación del personal que interactuará con el software, mediante el uso del manual de usuario. El Instituto de Suelos de Granma cuenta con los medios necesarios para la correcta implantación y ejecución del sistema. Existe una estrecha comunicación entre el desarrollador de la aplicación y los usuarios finales, logrando que el sistema que se desarrolle satisfaga las necesidades y expectativas de los clientes finales.

El sistema tendrá adjunto un respaldo de documentos que describen como está confeccionada la aplicación Web y un Manual de Usuario lo que garantiza cierto nivel de independencia para mantener el producto por los clientes. Los códigos están debidamente comentados, lo que facilita su entendimiento.

El sistema puede resultar vulnerable, desde el punto de vista de la ocurrencia de roturas o fallos del servidor, por lo que se recomienda realizar copias de seguridad tanto de la base de datos como de la aplicación en general. El sistema brindará un mensaje de alerta cada vez que el usuario va a realizar una acción de borrado o actualización de algún dato, dando la oportunidad de proseguir o cancelar la acción, esto trae consigo la no ocurrencia de errores humanos por descuido.

Por todo lo antes expuesto se determinó que es sostenible desde el punto de vista tecnológico el “Sistema de control de activos fijos tangibles del Instituto de Suelos de Granma”.

### **¿Es el sistema propuesto sostenible?**

Analizada la sostenibilidad del producto según las dimensiones administrativa, socio-humanista, ambiental y la tecnológica se puede afirmar de que la herramienta propuesta cumple con todos los requerimientos antes expuestos, por lo que se puede decir que constituirá un producto informático sostenible al ser usado por toda la comunidad, después que se finalice su desarrollo, la que se encargaría de su evolución.

## 2.4 Fase de Planificación

XP plantea la planificación como un permanente diálogo entre la parte empresarial y los programadores, en la que los primeros decidirán el alcance, ¿qué es lo realmente necesario del proyecto?, la prioridad qué debe ser hecha la historia de usuario, en primer lugar la composición de las versiones que debería incluir cada una de ellas y la fecha de las mismas. En cuanto a los programadores, son los responsables de realizar la estimación de la duración requerida para implementar las funcionalidades deseadas por el cliente, de informar sobre las consecuencias de determinadas decisiones, de organizar la cultura de trabajo y, finalmente, de realizar la planificación detallada dentro de cada versión (Escribano, 2007).

### 2.4.1 Estimación de esfuerzos por historias de usuarios

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas, permitiendo tener una medida real de la velocidad de progreso del proyecto y brindando una guía razonable a la cual ajustarse. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Esta fase dura unos pocos días.

Las estimaciones de esfuerzo asociado a la implementación de las historias la establecen los programadores utilizando como medida el punto. Un punto equivale a una semana ideal de programación, donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción. A continuación se muestra la tabla resumen de las historias de usuario donde se exponen los principales resultados según estos aspectos.

**Tabla 2.6** Estimación de esfuerzo por historia de usuario.

No	Historia de Usuario	Prioridad	Riesgo	Esfuerzo	Iteración
1	Autenticar usuario	Alta	Medio	0.1	1
2	Gestionar usuario	Alta	Medio	0.3	1
3	Gestionar privilegios	Alta	Medio	0.3	1

Capítulo 2. Descripción, construcción y valoración de la propuesta de solución

4	Asignar Usuario	Alta	Medio	0.1	1
5	Gestionar trabajador	Alta	Medio	0.3	1
6	Gestionar área	Alta	Medio	0.4	1
7	Gestionar local	Alta	Medio	0.3	1
8	Gestionar activos fijos tangibles	Alta	Alta	0.4	1
9	Gestionar control de servicios	Alta	Alta	0.6	1
10	Gestionar movimiento	Alta	Alta	0.6	2
11	Gestionar responsable de área	Medio	Medio	0.1	2
12	Gestionar responsable de local	Medio	Medio	0.1	2
13	Gestionar responsable de activos fijos tangibles	Medio	Medio	0.1	2
14	Acta de activos fijos tangibles.	Medio	Medio	0.2	3
15	Reporte de movimiento de medios básicos.	Medio	Medio	0.3	3
16	Reporte de servicio prestado a medios básicos.	Medio	Medio	0.2	3
17	Reporte de activos fijos tangibles.	Medio	Medio	0.3	3
18	Reporte de baja de activos fijos tangibles.	Medio	Medio	0.1	3
19	Reporte de activos fijos tangibles a almacenar.	Medio	Medio	0.2	3
20	Reporte de activos fijos tangibles tecnológico.	Medio	Medio	0.1	3
21	Salvar base de datos.	Medio	Alta	0.1	3
22	Restaurar base de datos.	Medio	Alta	0.1	3

### 2.4.2 Plan de duración de las iteraciones

Una vez identificadas las historias de usuario del sistema y estimado el esfuerzo dedicado a la realización de cada una de estas se crea el plan de duración de cada una de las iteraciones, Este plan se encarga de mostrar las historias de usuario que serán implementadas en cada una de las iteraciones, así como la duración estimada de cada una y el orden en que se implementarán.

De acuerdo a lo mencionado anteriormente se decidió realizar dicha planificación en tres iteraciones detalladas a continuación.

**Tabla 2.7** Plan de duración de las iteraciones.

<b>Iteraciones</b>	<b>Orden de las historias de usuario a implementar</b>	<b>Duración de las iteraciones</b>
Primera iteración	Autenticar usuario. Gestionar usuario. Asignar usuario. Gestionar privilegios. Gestionar trabajador. Gestionar área. Gestionar local. Gestionar activos fijos tangibles. Gestionar control de servicio.	4 semanas
Segunda iteración	Gestionar movimiento. Gestionar responsable de área. Gestionar responsable del local. Gestionar responsable de activos fijos tangibles.	1 semana y 2 días



Tercera iteración	<p>Acta de activos fijos tangibles.</p> <p>Reporte de movimiento de medios activos fijos tangibles.</p> <p>Reporte de servicio prestado a activos fijos tangibles.</p> <p>Reporte de activos fijos tangibles.</p> <p>Reporte de baja de activos fijos tangibles.</p> <p>Reporte de activos fijos tangibles a almacenar.</p> <p>Reporte de activos fijos tangibles tecnológico.</p> <p>Salvar base de datos.</p> <p>Restaurar base de datos.</p>	2 semanas y 2 días
-------------------	---	--------------------

### 2.4.3 Plan de entrega

Luego de determinar qué historias de usuario serán agrupadas para conformar una entrega, y el orden de las mismas se realiza el cronograma de entregas que establece la fecha acordada con el cliente para la liberación de las diferentes versiones. En la siguiente tabla se muestra el plan de duración de entregas en el cual se especifican un aproximado de las fechas para cada iteración.

**Tabla 2.8** Plan de entrega por iteraciones.

Iteración	Primera iteración	Segunda iteración	Tercera iteración
Entrega	Final 1ra Iteración 4ta semana de abril	Final 2da Iteración 1ra semana de mayo	Final 3ra Iteración 3ra semana de mayo

## 2.5 Iteraciones

Según la metodología XP esta es la fase principal en el ciclo de desarrollo, la implementación debe realizarse de forma iterativa, permitiendo así que al final de cada iteración surja un producto funcional que debe ser probado y mostrado al cliente, permitiendo de esta forma lograr una constante retroalimentación desarrolladores-cliente, este producto será previamente probado para incrementar la visión de los desarrolladores y el cliente de posibles cambios y soluciones. Las iteraciones son también utilizadas para medir el progreso del proyecto. Una iteración terminada sin errores es una medida clara de avance.

### 2.5.1 Tareas

En esta fase XP plantea la implementación de cada una de las historias de usuarios. Como parte de este plan se crean tareas para ayudar a organizar la implementación exitosa de las historias de usuarios (HU). En cada iteración XP propone la realización de varias tareas asignadas cada una a un programador específico. Estas historias de usuario son divididas en tareas de entre 1 y 6 días de duración.

A continuación se presentan algunas de las tablas de las tareas de usuario, se pueden consultar otras en el [Anexo 2](#).

**Tabla 2.9** Tarea insertar activos fijos tangibles.

Tarea	
<b>Número tarea:</b> T1H6	<b>Nombre de historia:</b> Gestionar activos fijos tangibles
<b>Nombre tarea:</b>	Insertar activos fijos tangibles.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 15/04/2013	<b>Fecha fin:</b> 15/04/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita ingresar los activos fijos tangibles y registrarlos en la base de datos del sistema.	

**Tabla 2.10** Tarea modificar activos fijos tangibles.

Tarea	
<b>Número tarea:</b> T2H6	<b>Nombre de historia:</b> Gestionar activos fijos tangibles.
<b>Nombre tarea:</b>	Modificar activos fijos tangibles.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 16/04/2013	<b>Fecha fin:</b> 16/04/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita modificar los activos fijos tangibles que están registrados en la base de datos del sistema.	

**Tabla 2.11** Tarea eliminar activos fijos tangibles.

Tarea	
<b>Número tarea:</b> T3H6	<b>Nombre de historia:</b> Gestionar activos fijos tangibles
<b>Nombre tarea:</b>	Eliminar activos fijos tangibles.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 17/04/2013	<b>Fecha fin:</b> 17/04/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita eliminar activos fijos tangibles de la base de datos del sistema.	

**Tabla 2.12** Tarea visualizar activos fijos tangibles

Tarea	
<b>Número tarea:</b> T4H6	<b>Nombre de historia:</b> Gestionar activos fijos tangibles
<b>Nombre tarea:</b>	Visualizar activos fijos tangibles.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 18/04/2013	<b>Fecha fin:</b> 18/04/2013

<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita visualizar los activos fijos tangibles que están registrados en la base de datos del sistema.

**Tabla 2.13** Tarea insertar área.

Tarea	
<b>Número tarea:</b> T1H4	<b>Nombre de historia:</b> Gestionar área
<b>Nombre tarea:</b>	Insertar área.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 7/04/2013	<b>Fecha fin:</b> 7/04/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita insertar las áreas en el sistema.	

**Tabla 2.14** Tarea modificar área.

Tarea	
<b>Número tarea:</b> T2H4	<b>Nombre de historia:</b> Gestionar área
<b>Nombre tarea:</b>	Modificar área.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 8/04/2013	<b>Fecha fin:</b> 8/04/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita al especialista económico del sistema modificar las áreas.	

Tabla 2.15 Tarea eliminar área.

Tarea	
<b>Número tarea:</b> T4H4	<b>Nombre de historia:</b> Gestionar área
<b>Nombre tarea:</b>	Eliminar área.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 9/04/2013	<b>Fecha fin:</b> 9/04/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita eliminar áreas de la base de datos del sistema.	

Tabla 2.16 Tarea visualizar área.

Tarea	
<b>Número tarea:</b> T5H4	<b>Nombre de historia:</b> Gestionar área
<b>Nombre tarea:</b>	Visualizar área.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 10/04/2013	<b>Fecha fin:</b> 10/04/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita visualizar las áreas que se encuentran almacenadas en la base de datos del sistema.	

## 2.6 Producción

En esta fase se realizan las tareas planificadas por iteración. Para ello se diseña, se codifica y se prueba. Al mismo tiempo, se deben tomar decisiones sobre la inclusión de nuevas características a la versión actual, debido a cambios durante esta fase. Las ideas que han sido propuestas y las sugerencias son documentadas para su posterior implementación.

### 2.6.1 Diseño

Este sistema fue diseñado para facilitar la gestión de los activos fijos tangibles del Instituto de Suelos de Granma de forma dinámica y agradable al usuario. Para lograrlo se emplearon algunos principios de diseño visual en las páginas web que la conforman.

El diseño visual define la apariencia del sistema y es de gran importancia para lograr que el usuario se sienta satisfecho con la información que obtiene y con la forma en que lo hace por eso la aplicación presenta un diseño simple y sencillo, sin muchas complicaciones, orientado al entorno de trabajo del cliente para que se sienta identificado con la aplicación.

Se eligieron los colores amarillo claro y carmelita claro y oscuro pues se considera que estos colores contribuyen a construir una interfaz agradable a la vista del usuario. Se utilizó el amarillo claro para el fondo de las páginas, el negro para las letras garantizando una lectura favorable de los textos. Se usan las letras Verdana, Arial, *Helvetica*, *sans-serif* para los textos de las páginas. Estos tipos de letra permiten una lectura rápida y cómoda. Es mínimo el uso de imágenes y animaciones para evitar largos tiempos de espera a la hora de cargar la página y visualizarlas.

Para la construcción del sistema se tomaron en cuenta algunos de los estándares de implementación propuestos: un *header*, donde se muestra la información general de sistema como logo de la institución, el nombre del sistema e imágenes y textos que muestren de manera general el contenido de la aplicación, un menú superior donde se encuentran los diferentes vínculos de acceso a las secciones del sistema, la sección del contenido donde se muestra la información que se desea buscar.

Algunas pantallas del sistema se pueden consultar en el [Anexos 3](#).

El diseño de la base de datos fue realizado con la herramienta *DBDesignerFork*, el mismo está compuesto por 20 tablas, la cuales están normalizadas, cumpliendo con las normas establecidas para el diseño de bases de datos. En el [Anexo 4](#) puede ser visto el modelo de datos de la aplicación.

### 2.6.2 Codificación

En la implementación del sistema se utiliza el lenguaje de programación web PHP (*Personal Home Page*), el cual es un lenguaje del lado del servidor y es diseñado originalmente para la creación de aplicaciones web dinámicas. Se emplean clases en el código fuente porque según las características del sistema se considera que es necesario utilizar la programación orientada a objetos. Además se hace evidente el uso de la reutilización de código pues las funcionalidades del sistema presentan características en común.

### 2.6.3 Pruebas

Uno de los pilares fundamentales de XP es el proceso de prueba, la cual constituyen el último bastión desde el que se puede evaluar la calidad de forma pragmática y descubrir los errores. Las pruebas son un conjunto de actividades que se pueden planificar por adelantado y llevar a cabo sistemáticamente. Por esta razón se debe definir en el proceso de la ingeniería del software. Todo esto contribuye a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones.

Las pruebas de aceptación son pruebas de caja negra que se realizan a partir de las historias de usuarios (Beck, 1999). Durante las iteraciones las historias de usuarios escogidas serán traducidas a prueba de aceptación. En ella se especifican, la perspectiva del cliente, y los escenarios para probar que la historia de usuario ha sido implementada correctamente. Una historia de usuario puede tener todas las pruebas de aceptación que desee para asegurar su funcionamiento. El objetivo específico de esta prueba es garantizar que los requerimientos han sido cumplidos y que el sistema ha sido aceptable. A continuación se muestran algunas de las pruebas de aceptación propuestas a realizarse, para las pruebas restantes consultar [Anexo 5](#).

**Tabla 2.17** Caso de prueba insertar correctamente activos fijos tangibles.

Caso de Prueba	
<b>Código de prueba:</b> H6_P1	<b>Número de historia de usuario:</b> 6
<b>Nombre de la prueba:</b> Insertar correctamente los activos fijos tangibles.	

<b>Descripción:</b> Prueba para verificar el ingreso correcto de un nuevo activos fijos tangibles en la base de datos. El Especialista económico accede a la opción de activos fijos tangibles, selecciona el área y local en el cual quiere agregar un activo fijo tangible, ingresa los nuevos activos fijos tangibles satisfactoriamente.
<b>Condiciones de ejecución:</b> El activos fijos tangibles no debe estar en la base de datos, loa datos ingresados deben estar correctos.
<b>Entrada:</b> Datos requeridos para el ingreso de los activos fijos tangibles válidos.
<b>Resultado esperado:</b> El activo fijo tangible se insertó correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

**Tabla 2.18** Caso de prueba error al insertar activos fijos tangibles.

Caso de Prueba	
<b>Código de prueba:</b> H6_P2	<b>Número de historia de usuario:</b> 6
<b>Nombre de la prueba:</b> Error al insertar activos fijos tangibles.	
<b>Descripción:</b> Prueba para verificar el error al ingreso de un nuevo activo fijo tangible dentro de una área y local del sistema. El Especialista económico accede a la opción de activo fijo tangible, selecciona el área y local en el cual quiere agregar un activo fijo tangible, ingresa los datos incorrectos, el sistema le indicara el error.	
<b>Condiciones de ejecución:</b> Entrada de datos incorrecta.	
<b>Entrada:</b> Datos requeridos para el ingreso de los activos fijos tangibles incorrectos.	
<b>Resultado esperado:</b> Mensaje indicando error en el ingreso de los datos.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Tabla 2.19** Caso de prueba modificar activos fijos tangibles.

Caso de Prueba	
<b>Código de prueba:</b> H6_P3	<b>Número de historia de usuario:</b> 6
<b>Nombre de la prueba:</b> Modificar correctamente activos fijos tangibles.	



<p><b>Descripción:</b> Prueba para verificar la modificación correcta de un activos fijos tangibles dentro de una área y local del sistema. El Especialista económico accede a la opción de activos fijos tangibles, modificar activos fijos tangibles, entrar los nuevos datos correctos y válidos, el sistema notifica: El activos fijos tangibles se modificó correctamente.</p>
<p><b>Condiciones de ejecución:</b> El activos fijos tangibles debe estar en la base de datos.</p>
<p><b>Entrada:</b> Datos requeridos para la modificación de activos fijos tangibles válidos.</p>
<p><b>Resultado esperado:</b> El local se modificó correctamente.</p>
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria.</p>

**Tabla 2.20** Caso de prueba error al modificar activos fijos tangibles.

Caso de Prueba	
<b>Código de Prueba:</b> H6_P4	<b>Número de Historia de Usuario:</b> 6
<b>Nombre de la Prueba:</b> Error al modificar un activos fijos tangibles.	
<p><b>Descripción:</b> Prueba para verificar errores en la modificación de los activos fijos tangibles dentro de una área y local del sistema. El Especialista económico accede a la opción de los activos fijos tangibles, modificar activos fijos tangibles, entrar los nuevos datos incorrectos, el sistema notifica: error al modificar el local.</p>	
<p><b>Condiciones de ejecución:</b> Que exista los activos fijos tangibles a modificar, datos a modificar ingresados incorrectamente.</p>	
<p><b>Entrada:</b> Datos requeridos para la modificación de los activos fijos tangibles incorrectos.</p>	
<p><b>Resultado esperado:</b> Mensaje indicando error en el ingreso de los datos.</p>	
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria.</p>	

**Tabla 2.21** Caso de prueba eliminar activos fijos tangibles correctamente.

Caso de Prueba	
<b>Código de Prueba:</b> H6_P5	<b>Número de Historia de Usuario:</b> 6
<b>Nombre de la Prueba:</b> Eliminar un activos fijos tangibles correctamente.	
<b>Descripción:</b> Prueba para verificar la eliminación de los activos fijos tangibles dentro de una área y local del sistema. El Especialista económico accede a la opción de activos fijos tangibles, eliminar activos fijos tangibles, selecciona activos fijos tangibles que desea eliminar, el sistema notifica: con un mensaje de confirmación para eliminar activos fijos tangibles, si lo confirma se eliminara los activos fijos tangibles seleccionados.	
<b>Condiciones de ejecución:</b> Que exista activos fijos tangibles a eliminar.	
<b>Entrada:</b> Se intenta eliminar activos fijos tangibles.	
<b>Resultado esperado:</b> Se elimina correctamente activos fijos tangibles.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Tabla 2.22** Caso de prueba error al eliminar activos fijos tangibles.

Caso de Prueba	
<b>Código de prueba:</b> H6_P6	<b>Número de historia de usuario:</b> 6
<b>Nombre de la prueba:</b> Error al eliminar activos fijos tangibles.	
<b>Descripción:</b> Prueba para verificar errores dentro de la eliminación de los activos fijos tangibles dentro de un área y local del sistema. El Especialista económico accede a la opción de activos fijos tangibles e intenta eliminar sin antes escoger los activos fijos tangibles.	
<b>Condiciones de ejecución:</b> Exista los activos fijos tangibles a eliminar.	
<b>Entrada:</b> Se accede a la opción de eliminar sin antes escoger activos fijos tangibles.	
<b>Resultado esperado:</b> Mensaje indicando que no se puede eliminar esos	

activos fijos tangibles.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

**Tabla 2.23** Caso de prueba insertar correctamente una área.

Caso de Prueba	
<b>Código de prueba:</b> H4_P1	<b>Número de historia de usuario:</b> 4
<b>Nombre de la prueba:</b> Insertar correctamente una área.	
<b>Descripción:</b> Prueba para verificar el ingreso correcto de una nueva área dentro del sistema. El especialista económico accede a la opción de área, insertar nueva área, ingresa los datos nuevos en la base de datos del sistema.	
<b>Condiciones de ejecución:</b> El área no debe estar en la base de datos, los datos deben ser correctos.	
<b>Entrada:</b> Datos requeridos para el ingreso del área válidos.	
<b>Resultado esperado:</b> El área se insertó correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Tabla 2.24** Caso de prueba error al insertar una área.

Caso de Prueba	
<b>Código de prueba:</b> H4_P2	<b>Número de historia de usuario:</b> 4
<b>Nombre de la prueba:</b> Error al insertar correctamente un área.	
<b>Descripción:</b> Prueba para verificar el error al ingresar una nueva área dentro del sistema. El especialista económico accede a la opción de área, insertar nueva área, ingresa algún dato incorrecto, el sistema notifica: error al ingresar la nueva área.	

<b>Condiciones de ejecución:</b> Datos ingresados erróneamente.
<b>Entrada:</b> Datos requeridos para el ingreso del área no válidos.
<b>Resultado esperado:</b> El área no se puede insertar en el sistema, se notifica: error al ingresar la nueva área.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

**Tabla 2.25** Caso de prueba modificar correctamente una área.

Caso de Prueba	
<b>Código de prueba:</b> H4_P3	<b>Número de historia de usuario:</b> 4
<b>Nombre de la prueba:</b> Modificar correctamente una área.	
<b>Descripción:</b> Prueba para verificar la modificación correcta de una área dentro del sistema. El especialista económico accede a la opción de área, modificar área, entrar los nuevos datos correctos y válidos, los datos se modifican en la base de datos.	
<b>Condiciones de ejecución:</b> El área a modificar debe estar en la base de datos, los nuevos datos ingresados deben ser correctos.	
<b>Entrada:</b> Datos requeridos para la modificación del área válidos.	
<b>Resultado esperado:</b> El área se modifica correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Tabla 2.26** Caso de prueba error al modificar una área.

Caso de Prueba	
<b>Código de prueba:</b> H4_P4	<b>Número de historia de usuario:</b> 4
<b>Nombre de la prueba:</b> Error al modificar una área.	
<b>Descripción:</b> Prueba para verificar errores en la modificación de una área dentro del sistema. El especialista económico accede a la opción de área, modificar área, entrar los nuevos datos, el sistema notifica: error al modificar el área.	

<b>Condiciones de ejecución:</b> El área a modificar debe estar en la base de datos y los nuevos datos ingresados deben ser incorrectos.
<b>Entrada:</b> Datos requeridos para la modificación del área no válidos.
<b>Resultado esperado:</b> El sistema notifica: error al modificar el área.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

**Tabla 2.27** Caso de prueba eliminar una área correctamente.

Caso de Prueba	
<b>Código de prueba:</b> H4_P5	<b>Número de historia de usuario:</b> 4
<b>Nombre de la prueba:</b> Eliminar un área correctamente.	
<b>Descripción:</b> Prueba para verificar la eliminación de una área dentro del sistema. El especialista económico accede a la opción de área, eliminar área, selecciona el área que desea eliminar, el sistema emite un mensaje de confirmación para eliminar una área, si lo confirma se eliminará el área seleccionada.	
<b>Condiciones de ejecución:</b> El área debe estar en la base de datos	
<b>Entrada:</b> Se intenta eliminar un área.	
<b>Resultado esperado:</b> El área es eliminada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Tabla 2.28** Caso de prueba error al eliminar una área.

Caso de Prueba	
<b>Código de prueba:</b> H4_P6	<b>Número de historia de usuario:</b> 4
<b>Nombre de la Prueba:</b> Error al eliminar un área.	
<b>Descripción:</b> Prueba para verificar errores dentro de la eliminación de un área del sistema. El especialista económico accede a la opción de área, eliminar área e intenta eliminar área sin seleccionarla.	

<b>Condiciones de ejecución:</b> El área debe estar en la base de datos
<b>Entrada:</b> Se intenta eliminar un área sin ser seleccionada previamente.
<b>Resultado esperado:</b> El sistema emite mensaje de error.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

**Tabla 2.29** Caso de prueba visualizar áreas correctamente.

Caso de Prueba	
<b>Código de prueba:</b> H4_P7	<b>Número de historia de usuario:</b> 4
<b>Nombre de la prueba:</b> Visualizar áreas correctamente.	
<b>Descripción:</b> Prueba para verificar si se pueden visualizar los datos de las áreas dentro del sistema. El especialista económico accede a la opción de área, visualizar área, el sistema muestra la áreas que contiene la base de datos.	
<b>Condiciones de ejecución:</b> Deben existir áreas en la base de datos.	
<b>Entrada:</b> Se intenta visualizar áreas en el sistema.	
<b>Resultado esperado:</b> Datos de las áreas visualizados.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

## 2.7 Mantenimiento

Mientras la primera versión se encuentra en producción, el proyecto XP mantiene el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para esto se realizan tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción. Se puede requerir además de nuevo personal dentro del equipo y cambios en su estructura.

Durante el desarrollo del sistema se tuvo en cuenta las opiniones del cliente para el cual fue implementado, de esta forma en esta etapa fueron identificadas cada una de las nuevas necesidades del usuario que pudieron generar cambios en el software.

### **2.8 Muerte del proyecto**

Se produjo cuando el cliente no tuvo más historias de usuario para ser incluidas en el sistema. El cliente quedó satisfecho en todos los aspectos como rendimiento y confiabilidad del sistema. A partir de aquí se generó la documentación final del sistema y no se realizaron más cambios en la arquitectura. Además el sistema implementado tiene buen rendimiento.

### **2.9 Conclusiones del capítulo**

El análisis del funcionamiento del sistema de control de activos fijos tangibles del Instituto de Suelos de Granma permitió definir las características necesarias para la creación de una aplicación web acorde a las peticiones del cliente, todo esto a través de la metodología de desarrollo software escogida. Es así como en este capítulo se detallan los resultados de cada una de las fases que XP propone. Se construyó la aplicación Web que introdujo una nueva vía para gestionar la información de activos fijos tangibles en la entidad, lo cual contribuyó a solucionar los problemas que originaron esta investigación.

### **Conclusiones generales**

- Se caracterizó el proceso de gestión de la información de los activos fijos tangibles del Instituto de Suelos de Granma, logrando una mejor comprensión del tema, lo que facilitó la búsqueda de la solución a la problemática planteada.
- Se desarrolló un sistema informático basado en la web para la gestión de la información de los activos fijos tangibles del Instituto de Suelos de Granma, que centraliza la dicha información y garantiza una mejor seguridad para el acceso a la misma.



### **Recomendaciones**

- Generalizar el sistema informático resultado de la presente investigación a otras instituciones del país.
- Realizar investigaciones futuras para integrar el sistema propuesto con el “Sistema de Control de Asistencia de Personal y Recursos Humanos” actualmente usado en la institución.
- Utilizar la presente investigación como material de estudio para la realización de trabajos similares.

---

**Referencias Bibliográficas**

- ASKALL ORG, S. A. (2007). *Programando en Java*. Standfor: Addison-Wesley Company.
- ASSETS (2006). ASSETS: Sistema de Gestión Integral Retrieved 13-12-2010, 2010, from <http://www.assets.co.cu/>
- Ávila A & Pérez B (2009). *Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas. Unpublished Pre grado*. Univercidad de las Ciencias Informáticas, Ciudad de La Habana.
- Beck, K. (1999). *Extreme Programming XP. Embrace Change*: Pearson Education.
- Bellman, R. E. (2007). *Web Programming* (Vol. 2). Princenton Princenton University Press.
- Bowden, D. (2007). Rational Rose Tutorials Retrieved 12-1-2010, 2010, from <http://homepages.uel.ac.uk/D.Bowden/>
- Brusilovsky, P., Gilles, G., Frasson, C., y VanLehn, K. (2007). *Course sequencing for Static Courses? Applying ITS Techniques in Large-Scale Web\_Based Education* (Springer Verlag ed.): Proceedings of Intelligent Tutoring Systems.
- Cano Hidalgo, I. (2006). Sistema de gestión Integral ASSETS-NS: garantía de seguridad total Retrieved 12-12-2010, 2010, from [http://www.cadenagramonte.cubaweb.cu/ciencia/sistemas\\_gestion\\_integ ral.asp](http://www.cadenagramonte.cubaweb.cu/ciencia/sistemas_gestion_integ ral.asp)
- Cinca, T. C. (2008). Características principales de la Web.Web Estática y Web Dinámica. *Aplicaciones para la gestión*, 12(4), 9-14.
- Cobo, Á., Gómez, P., Pérez, D., y Rocha, R. (Eds.). (2005). *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web* (Vol. I). Madrid, España: Diaz de Santos.
- Colombo, B., Demetrescu, C., Finocchi, I., y Laura, L. (2003). A Java-based System for Building Animated Presentations over the Web. *Elsevier Science of Computer Programing*, 901-946.
- Conallen, J. (2006). *Building Web Applications with UML*.
- Concepción García, M. R. (2000). *La gestión ambiental de proyectos informáticos sostenibles por estudiantes de Ingeniería Informática*.

- Paper presented at the La gestión ambiental de proyectos informáticos sostenibles por estudiantes de Ingeniería Informática.
- de Armas Phillips, J. R., y Machado Barrios, M. d. I. Á. (2010). CONTROL DE LOS ARCHIVOS FIJOS TANGIBLES PARA FOMENTAR UNA CULTURA ECONÓMICA EN DIRECTIVOS, TRABAJADORES Y ESTUDIANTES EN LA UNIVERSIDAD DE CIENCIAS PEDAGÓGICAS “Enrique José Varona”. *Revista IPLAC*, 4, 15.
- Escribano, G. F. (2007). *Introducción a Extreme Programming XP. A gentle introduction Retrieved 03/03: 2012.*
- Frentzen, J., y Henry, S. (Eds.). (1999). *Superutilidades para JavaScript*. Madrid, España: McGraw-Hill, Inc.
- Furgeta, A. (2002). *Introduction to Systems Engineering*. New York: MIT Press.
- Gilfillan, I. (2004). *La biblia de MySQL* (Anaya Multimedia ed.).
- Gilmore, W. J. (2006). *Beginning PHP and MySQL 5 From Novice to Professional, Second Edition* (Segunda ed.). Estados Unidos.
- Hernán Ruiz, M. (2006). *Programación Web avanzada: Soluciones rápidas y efectivas para desarrolladores de sitios*. La Habana: Editorial Félix Varela.
- Hernández, R., y Greguas, D. (2010). Estándares de Diseño Web. *Ciencias de la Información*, 41, No.2.
- Herrera, F., Sandoval, J., y Quijada, J. (2009). Incorporación de nuevas tecnologías Retrieved 1-12-2010, 2010, from <http://WWW.rediris.es/rediris/boletín/41-42/ponencia13.html>
- I.M. Lockhart (2009). *Gestor de Base de Datos Postgre SQL*.
- Jacobson, I., Booch, G., y Rumbaugh, J. (2000). *El Proceso Unificado de Desarrollo de Software* (Addison-Wesley Company ed.). Stanford.
- JBC (2009). *Manual de JBC (Java Byte Code)*.
- JDK (2009). Microsystem del JDK 1.0 (Java Development Kit).
- Lockhart, T. (2009). *Tutorial de PostgreSQL. 111-116*
- Lynch, E. (2003). Constructivism and Distance Education. Internet Retrieved 12-6-2010, 2010, from

<http://seamonkey.ed.asu.edu/~mcisaac/emc703old97/spring97/7/lynch7.htm>

- Málaga, D. (2008). *Manual de HTML y JavaScript*.
- Mora Guevara, D. d. I. C., y López Escala, Y. (2012). *Sistema para la gestión de recursos materiales y contratos en la Unidad Empresarial Básica Diseño y Servicios de Ingeniería de Granma Universidad de Granma*. Bayamo.
- Oliver, K. (2000). Methods for developing constructivist learning on the web. *Educational Technology Research & Development*, 6, 1-18.
- Prentzas, J., Hatzilygeroudis, I., y Garofalakis, J. (2002). A web-Based Intelligent Tutoring System Using Hybrid Rules. *Intelligent Tutoring Systems- ITS 2002*, 119-128.
- Pressman, R. S. (1996). *Ingeniería de Software: Un Enfoque Práctico* (Addison-Wesley Company ed. Vol. 1). Standford.
- Sanz, A. (2002). Introducción a UML Retrieved 22-04-2006, 2006, from <http://www.yoprogramo.com/docs/umlintro.doc>.
- SQL (2007). *Mysql Retrieved 11-1-2011*.
- Valles, J. G. (2008). El modelo Cliente-Servidor. 23-34.
- Villa, L. (2006). La importancia creciente de la accesibilidad de sitios web Retrieved 11-1-2011, 2011, from <http://www.alzado.org/articulo.php>
- Zelkowitz, M. V., Shaw, A. C., y Gannon, J. D. (2009). *Principles of Software Engineering and Design* (MIT Press ed. Vol. 2).

## Anexos

### Anexo 1 Historias de usuario

Historia de usuario “Gestionar responsable de local”.

Historia de Usuario	
Número: 13	Usuario: Especialista económico y jefe de área
Nombre de historia: Gestionar responsable de local	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración asignada: 2
Programador responsable: Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
Descripción: Permite asignar, visualizar y actualizar un trabajador o usuario a cargo de un local dentro de una área.	
Observaciones:	

Historia de usuario “Gestionar responsable de activos fijos tangibles”.

Historia de Usuario	
Número: 14	Usuario: Especialista económico y jefe de área
Nombre de historia: Gestionar responsable de activos fijos tangibles.	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Puntos estimados: 0.1	Iteración asignada: 2
Programador responsable: Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
Descripción: Permite asignar, visualizar y actualizar un trabajador o usuario como el responsable de los activos fijos tangibles de un local o área.	
Observaciones:	

Historia de usuario "Reporte de servicios prestados a activos fijos tangibles".

Historia de Usuario	
Número: 17	Usuario: Especialista económico y jefe de área
Nombre de historia: Reporte de servicio prestado a los activos fijos tangibles.	
Prioridad en negocio: Medio	Riesgo en desarrollo: Medio
Puntos estimados: 0.2	Iteración asignada: 3
Programador responsable: Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo.	
Descripción: Permite visualizar informe donde se muestre el número de veces que se ha realizado un servicio prestado a un activo fijo tangible, ya sea mantenimiento a los activos fijos tangibles físicos o tecnológicos.	
Observaciones:	

## Anexo 2 Tareas

Tarea “Interfaz gestionar responsable de local”.

Tarea	
<b>Número tarea:</b> T1H13	<b>Nombre de historia:</b> Gestionar Responsable de Local
<b>Nombre tarea:</b>	Interfaz gestionar responsable de local.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0.1
<b>Fecha inicio:</b> 5/05/2013	<b>Fecha fin:</b> 5/05/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñarán e implementarán interfaces que permitan asignar, visualizar y actualizar un trabajador o usuario a cargo de un local dentro de un área.	

Tarea “Interfaz gestionar responsable de activos fijos tangibles”.

Tarea	
<b>Número tarea:</b> T1H14	<b>Nombre de historia:</b> Gestionar responsable de activos fijos tangibles
<b>Nombre tarea:</b>	Interfaz gestionar responsable de activos fijos tangibles.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0,1
<b>Fecha inicio:</b> 6/05/2013	<b>Fecha fin:</b> 6/05/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñarán e implementarán interfaces que permitan asignar, visualizar y actualizar un trabajador o usuario como el responsable de los activos fijos tangibles de un local o área.	

Tarea “Interfaz reporte de servicios prestados a los activos fijos tangibles”.

Tarea	
<b>Número tarea:</b> T1H17	<b>Nombre de historia:</b> Reporte de servicio prestado a los activos fijos tangibles
<b>Nombre tarea:</b>	Interfaz reporte de servicios prestados a los activos fijos tangibles.
<b>Tipo de tarea:</b> Diseño - Implementación	<b>Puntos estimados:</b> 0.2
<b>Fecha inicio:</b> 12/05/2013	<b>Fecha fin:</b> 13/05/2013
<b>Programador responsable:</b> Eddison Tomás Lemas Chuqui, Marco Vinicio Naranjo Gallo	
<b>Descripción:</b> Se diseñará e implementará una interfaz que permita visualizar en un reporte el número de veces que se ha realizado un servicio prestado a los activos fijos tangibles, ya sea mantenimiento a los activos fijos tangibles físicos o tecnológicos.	



## Anexo 3 Pantallas representativas del sistema.

### Pantalla de acceso al sistema.

Instituto de Suelos

Lunes, 10 de Junio del 2013

**Activos Fijos Tangibles**  
Sistema de Gestión

Si es usuario de este sistema deberá introducir sus datos para acceder al mismo.

**Autenticación**

Usuario: marco

Contraseña: ●●●●

Entrar

Bienvenido al Sistema de Gestión de Información para el Control de Medios Básicos del Instituto de Suelos de Granma.

El sistema brinda un conjunto de utilidades para la gestión y control de los medios básicos del Instituto de Suelos, aunque por sus características puede ser extendido a otras áreas sin modificación alguna. La interfaz de usuario lo hace un software de fácil manejo por lo que no se requiere de grandes conocimientos de informática, puede ser manejado con facilidad por personal no especialista sin dificultad alguna.

Este sistema ha sido diseñado e implementado solo para uso del Instituto de Suelos de Granma, el autor no se hace responsable de la pérdida o daño de la información por uso indebido del mismo.

Facultad de Ciencias Técnicas | Universidad de Granma  
© Copyright 2013 | Todos los derechos reservados

**Activos Fijos Tangibles**

## Pantalla de visualizar activos fijos tangibles

Instituto de Suelos

**Activos Fijos Tangibles**  
Sistema de Gestión

Activos Fijos Tangibles

Lunes, 10 de Junio del 2013 Usuario activo: marco

Inicio Administración Configuración **Activos fijos tangibles** Reportes Ayuda marco

**Visualizar Activos Fijos Tangibles**

Seleccione el Área: Economía  
Seleccione el Local: Caja de pago

Mostrar: 10      Insertar nuevo activo fijo tangible | Generar Reporte

Número Inventario	Código	Nombre	Valor M.N.	Valor C.U.C	Estado	Degreciación Anual	Fecha de Explotación	Marca	Modelo	Opciones
156	234	MOUSE	10.00	10.00	Bueno	5.00	2013-06-06	genius	x110 scroll	X
321	123	CPU	0.00	100.00	Bueno	10.00	2011-11-10	ASUS	A-123	X
376	113	MESA	67.00	67.00	Regular	54.00	2013-06-06	---	---	X
46478	989	TECLADO	0.00	22.00	Regular	0.50	2011-11-10	Aiteg	A-0012	X

Mostrando **Activos Fijos Tangibles** desde el **1** hasta el **4** de un total de **4**

Facultad de Ciencias Técnicas | Universidad de Gramina  
© Copyright 2013 | Todos los derechos reservados

**Activos Fijos Tangibles**

## Pantalla de insertar activos fijos tangibles.

Instituto de Suelos

Activos Fijos Tangibles Sistema de Gestión

Activos Fijos Tangibles

Lunes, 10 de Junio del 2013 Usuario activo: marco

[Inicio](#) [Administración](#) [Configuración](#) [Activos fijos tangibles](#) [Reportes](#) [Ayuda](#) [marco](#)

Ir a visualizar activo fijo tangible

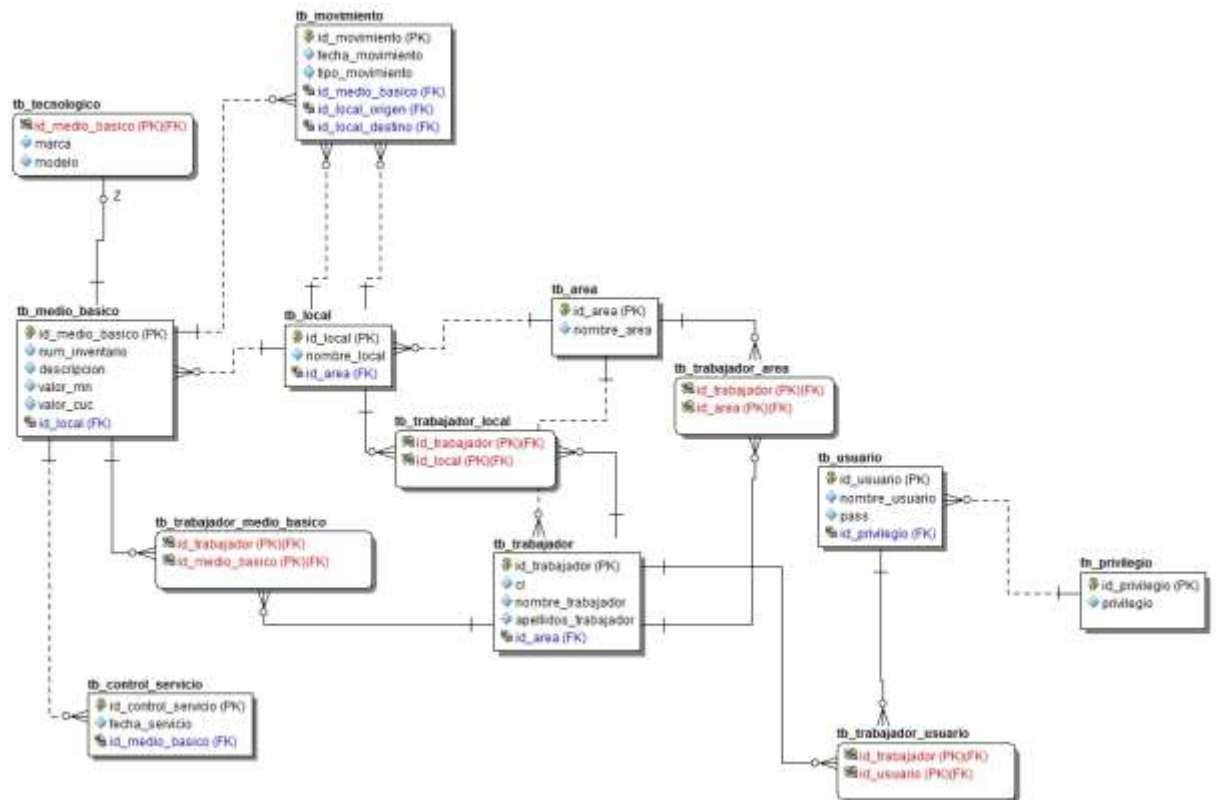
Los campos señalados (\*) son obligatorios.

Entrada de Medios Básicos:	
Área:	Economía *
Local:	Caja de pago *
Número de inventari:	1237034 *
Nombre:	Monitor LCD *
Código:	125
Valor M.N.:	150.00
Valor C.U.C.:	150.00
Estado:	Bue
Depreciación Anual:	5.00
Fecha de Explotación:	2013-04-05
Tecnológico:	SI
Marcas:	LG
Modelo:	Flatron
<input type="button" value="Entrar"/>	

Facultad de Ciencias Técnicas | Universidad de Gramma  
© Copyright 2013 | Todos los derechos reservados

Activos Fijos Tangibles

Anexo 4 Diseño de la base de datos



## Anexo 5 Pruebas

Caso de prueba “Error al asignar responsable de local”.

Caso de Prueba	
<b>Código de prueba:</b> H13_P2	<b>Número de historia de usuario:</b> 13
<b>Nombre de la prueba:</b> Error al asignar responsable de local.	
<b>Descripción:</b> Prueba para verificar errores en la asignación de un responsable de local dentro del sistema. El usuario accede a la opción de configuración, local, asignar responsable, el sistema mostrará un formulario para introducir los datos del usuario. Si no son válidos no se asignará el responsable y emitirá mensaje indicando el error.	
<b>Condiciones de ejecución:</b> Se debe tener ingresado trabajadores para realizar la asignación del usuario.	
<b>Entrada:</b> Datos del usuario correctos.	
<b>Resultado esperado:</b> No se asigna el responsable y emite mensaje indicando el error.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Caso de prueba “Asignación correcta de responsable de activo fijo tangible”.

Caso de Prueba	
<b>Código de prueba:</b> H14_P1	<b>Número de historia de usuario:</b> 14
<b>Nombre de la prueba:</b> Asignación correcta de responsable de activos fijos tangibles.	
<b>Descripción:</b> Prueba para verificar la asignación correcta de un responsable de activos fijos tangibles dentro del sistema. El usuario accede a la opción de configuración, activos fijos tangibles, asignar responsable, el sistema pedirá los datos del usuario, si son válidos se asignará el responsable correctamente.	
<b>Condiciones de ejecución:</b> Se debe tener ingresado trabajadores para realizar la	

asignación del usuario.
<b>Entrada:</b> Datos de usuario correctos.
<b>Resultado esperado:</b> Responsable asignado correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Caso de prueba “Visualización correcta del reporte de servicios prestados a los activos fijos tangibles”.

Caso de Prueba	
<b>Código de Prueba:</b> H17_P1	<b>Número de Historia de Usuario:</b> 17
<b>Nombre de la Prueba:</b> Visualización correcta del reporte de servicios prestados a los activos fijos tangibles.	
<b>Descripción:</b> Prueba para verificar la visualización correcta del reporte de servicios prestados de activos fijos tangibles. El usuario accede a la opción de generar reporte de servicios prestados de activos fijos tangibles y el sistema visualiza un informe con el número de veces que se ha realizado un servicio prestado de activos fijos tangibles.	
<b>Condiciones de ejecución:</b> Deben existir servicios prestados a los activos fijos tangibles en la base de datos.	
<b>Entrada:</b> Se accede a la opción de generar reporte de servicios prestados a los activos fijos tangibles.	
<b>Resultado esperado:</b> Se genera correctamente el reporte de servicios prestados a los activos fijos tangibles.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	